BIOLOGICALLY INSPIRED HETEROGENEOUS MULTI-AGENT SYSTEMS

A Thesis
Presented to
The Academic Faculty

by

Musad A. Haque

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2010

BIOLOGICALLY INSPIRED HETEROGENEOUS MULTI-AGENT SYSTEMS

Approved by:

Professor Magnus Egerstedt, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor David Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Ayanna Howard
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Tucker Balch
College of Computing
Georgia Institute of Technology

Professor Patricio Vela
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: 12 November 2010

To Ammu and Baba.

ACKNOWLEDGEMENTS

I thank Dr. Magnus Egerstedt, my advisor, for allowing me to develop this unorthodox research project and for his willingness to take a chance with me. I sincerely appreciate his words of encouragement, infectious enthusiasm, and support throughout the entire process (which also includes the maximum allowable number of prelim attempts).

I would like to thank the members of my dissertation committee: Drs. Ayanna Howard, Tucker Balch, David Anderson, and Patricio Vela for their many helpful suggestions and support. I am grateful to Drs. Lori Marino, Michael Goodisman, and Richard Connor for their helpful discussions on the biological aspect of my research. Special thanks go to Drs. Clyde Martin, Anthony Yezzi, and Giuseppe Notarstefano for their collaboration on various parts of the project.

I was fortunate to work with a great group of people at the GRITS lab and I appreciate their support and will cherish their friendships. I thank all my former colleagues: Drs. David Wooden, Tejas Mehta, Florent Delmotte, Meng Ji, Brian Smith, Dennis Ding, and Patrick Martin. And I wish the best of luck to all my current colleagues: Jean-Pierre de la Croix, Jonghoek Kim, Peter Kingston, Rahul Chipalkatty, Philip Twu, Amy LaViers, Hassan Jaleel, Waseem Abbas, and Greg Droge. A very special thanks goes to Dr. Amir Rahmani, with whom I've had the pleasure of working with closely on numerous components of this work. His collaboration has further improved my work.

I am thankful to my loving fiancèe, Shama, for being there by my side during this journey. She has an ability to simplify things, and this made the journey a lot easier.

Finally, I am grateful to my loving parents, Khurshid Jahan and Mahfuz Haque,

for their unwavering faith that I could do anything. I cannot thank them enough for making sacrifices so that I could pursue my dreams. This dissertation is dedicated to them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Many biological systems are known to accomplish complex tasks in a decentralized, robust, and scalable manner - characteristics that are desirable to the coordination of engineered systems as well. Inspired by nature, we produce coordination strategies for a network of heterogenous agents and in particular, we focus on intelligent collective systems. Bottlenose dolphins and African lions are examples of intelligent collective systems since they exhibit sophisticated social behaviors and effortlessly transition between functionalities. Through preferred associations, specialized roles, and self-organization, these systems forage prey, form alliances, and maintain sustainable group sizes. In this thesis, we take a three-phased approach to bioinspiration: in the first phase, we produce agent-based models of specific social behaviors observed in nature. The goal of these models is to capture the underlying biological phenomenon, yet remain simple so that the models are amenable to analysis. In the second phase, we produce bio-inspired algorithms that are based on the simple biological models produced in the first phase. Moreover, these algorithms are developed in the context of specific coordination tasks, e.g., the multi-agent foraging task. In the final phase of this work, we tailor these algorithms to produce coordination strategies that are ready to be deployed in target applications.

CHAPTER I

INTRODUCTION

In nature, there are numerous examples of systems that accomplish complex tasks in a decentralized, robust, and scalable manner [27]. For example, consider the schooling behavior of fish, where the coordinated movement of the group emerges from local interactions between individuals. Schools are usually formed to avoid predation and they operate without the presence of a leader. More interestingly, a collection of sub-tasks, such as alignment and separation with neighbors, allows for a complicated maneuver like bifurcation to take place near obstacles. Decentralized, robust, and scalable coordinations are characteristics desirable to engineered systems as well. Consider a network with a central decision-making agent. If that agent fails, it jeopardizes the completion of a task by the remaining functioning agents. Moreover, robustness maintains functionality of a network despite the loss of *any* agent in the network, and scalability prevents the coordination strategy from breaking down by the addition of one more agent to the network. As such, the purpose of this research is to develop biologically inspired coordination strategies for engineered systems, such as a network of unmanned vehicles.

The use of such vehicles has greatly increased and at present, objectives for these vehicles range from surveillance in counter-piracy operations to actual combat in counter-insurgency missions [2]. These vehicles can change missions in mid-flight, making them highly adaptive, and can fly for long hours in a mission as they are not constrained by the limitations of a human pilot. For example, the US Navy predicts that future combat operations will include teams of unmanned vehicles - light combat ships (LCS), unmanned aerial vehicles (UAVs), and unmanned underwater vehicles

(UUVs) - coordinating with each other, autonomously, in time-critical missions [2].

It is imperative for coordination strategies during cooperative (and competitive) tasks involving such a heterogeneous network of autonomous agents to address issues like assignment, distribution, and scheduling of tasks. Through proper assignment rules, the coordination strategy can utilize the different capabilities of the vehicles. A surveillance vehicle being assigned to a combat role is an example of a poor strategy. An example of inefficient distribution would be a case in which multiple combat vehicles, where the capabilities of the vehicles are not complementary, approach the same target. Also, consider the scenario where a team of vehicles is assigned to perform a task and each vehicle in the team is responsible for a certain subtask. For time-critical missions, it is important that the coordination strategy efficiently schedules these subtasks.

Immediately, we notice that *homogenous* organisms, such as fish - with identical agent types, collective decision-making, and reactive behaviors - fall short of the inspiration needed to design the desired coordination strategies. To this end, we turn our attention to more intelligent collective systems. In particular, we focus on Bottlenose dolphins and African lions, biological systems that exhibit sophisticated social behaviors and effortlessly transition between functionalities [16, 18, 50]. Through preferred associations, specialized roles, and self-organization, these systems forage prey, form alliances, and maintain sustainable aggregations (for details, see [16, 23]). Similar to the previous example of schooling, these systems successfully perform tasks, in a dynamic environment, through decentralized decision-making and the dissemination of only local information. In this thesis, we will allow these systems to guide us in the design of bio-inspired coordination strategies. In fact, we take the following *three-phased approach* to bioinspiration:

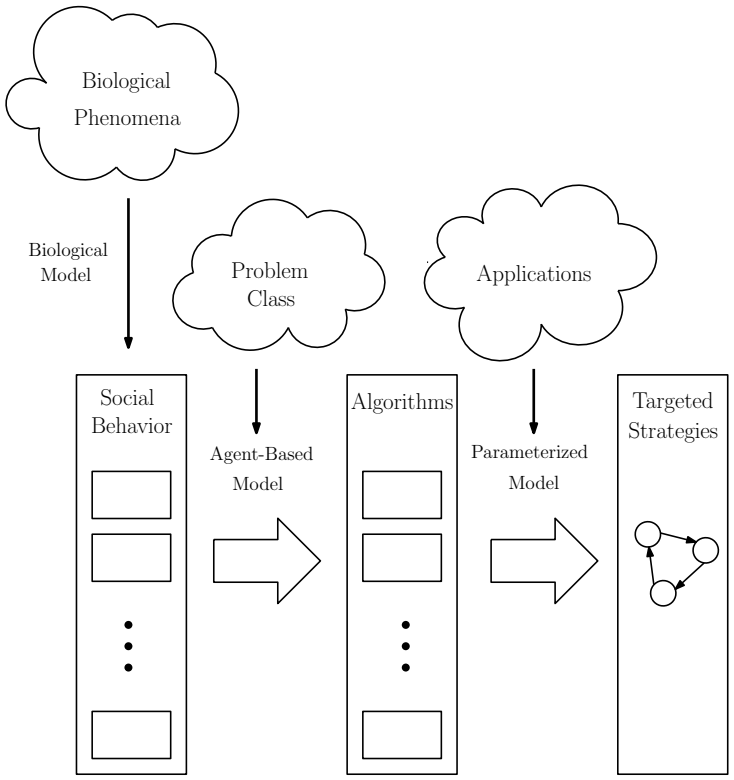


Figure 1: The approach to bioinspiration is shown. This work produces coordination strategies for target applications based on the social behavior of natural systems.

1. **Produce an agent-based model,**

2. **Produce tunable algorithms, and**

3. **Develop targeted coordination strategies.**

Figure 1 illustrates how biologically inspired coordination strategies are developed in this thesis. The design of a coordination strategy, for a specific application, begins with an observed social behavior. This behavior is characteristic to a certain biological model, which in turn, belongs to a class of biological phenomena. For example, consider the social behavior of forming alliances. This behavior is observed in Bottlenose dolphins, which is the biological model in this example, and the model belongs to the self-organizing class of biological phenomena.

Thus, from a list of interesting biological phenomena, such as predation and aggregation, we have a catalogue of social behaviors and in the first phase, we produce agent-based models of these social behaviors. With potential engineering applications at the back-end of this work, the goal of this phase is to produce simple, yet expressive models. Thus, we need to address how to capture the underlying biological phenomena using a simple model that can be easily be implemented in engineered devices.

In Figure 1, we illustrate the progression from a catalogue of social behaviors to a library of tunable algorithms. These algorithms are based on agent-based models (the outcome of the first phase), but are driven by the choice of a particular problem from the problem class. The multi-agent foraging task is an example of the types of problems addressed in this thesis and consequently, some of the challenges in this phase include the design of efficient foraging strategies. In developing the algorithms in this phase, a set of parameters are introduced, which are no longer tied to the description of the natural system; instead, these "tweak-able" parameters relate to the specific problem being addressed. As a result, when an algorithm is selected from this library, the outcome is a parameterized model of the social behavior, as shown in Figure 1.

In the final phase, the goal is to produce targeted strategies, based on the parameterized models, for engineered applications. The key challenge of this phase is to tailor the bio-inspired algorithms developed in the second phase so that they are ready to be deployed in target applications.

Based on this approach to bioinspiration, the work presented in this thesis is classified into three parts:

- Part I. **Models -** Chapters 3, 4, and 5

  In this part, we produce agent-based models, where each model represents an outcome of selecting a social behavior from the catalogue of behaviors (Figure

1).

- Part II. **Algorithms -** Chapters 6 and 7

  In this part of the thesis, we produce a library of tunable algorithms based on the models developed in Part I. The specific problems addressed in this part are the confinement problem and the multi-agent foraging task.

- Part III. **Applications -** Chapter 8

  In this part, we develop a coordination strategy for a network of unmanned vehicles for a specific engineered application. The strategy is developed according to the progression shown in Figure 1.

The remainder of the thesis is organized as follows: Chapter 2 presents background research on multi-agent systems and models of large-scale biological systems, and discusses the observed social behaviors of Bottlenose dolphins and African lions. In Chapters 3-5, we produce agent-based models of the following social behaviors: alliance forming of male Bottlenose dolphins, foraging method of Bottlenose dolphins, and the social structures of African lion prides. In Chapters 6-7, we develop bio-inspired algorithms for the following coordination problems: the confinement of non-cooperative agents and the multi-agent foraging task. These algorithms build on the model of foraging dolphins provided in Chapter 4. In Chapter 8, we produce targeted strategies for specific engineered applications. More specifically, we consider a search-and-destroy mission known as the suppression of enemy air defenses. Finally, concluding remarks are presented in Chapter 9.

CHAPTER II

BACKGROUND

In this chapter, we present previous works on biologically inspired multi-agent systems. These works have focused on the collective behavior of social insects and fish. Moreover, we refer to these systems as large-scale biological systems due to the size of their aggregations (e.g., the army ant *Eciton burchelli* studied in [18] can contain up to $200,000$ foragers in a colony.) The coordination strategies developed in this thesis focus on small-scale biological systems (primarily, dolphins and lions), which tend to organize in smaller groups and exhibit more complex interactions. As such, in this chapter, we also discuss previous works on social behaviors of Bottlenose dolphins and African lions observed in nature. Finally, since the bio-inspired coordination strategies are presented in the context of multi-agent systems, we provide recent results from this field as well.

2.1 Agent-Based Models of Large-Scale Biological Systems

An agent-based model is a representation of a system as a collection of decision-making units called agents. Agent-based models of biological systems have been studied extensively (see [8, 18, 19, 30, 40, 50, 61, 65, 66, 79] for a representative sample). The biological systems that have been examined are large-scale systems, most notably social insects (ants and bees) and fish.

Aggregations of animals were simulated in [79]. There, the flocking behavior of was simulated by incorporating behaviors - avoid collisions, match velocities, and remain close to nearby flock mates - into the agents as opposed to scripting their paths. Moreover, the computational advantages of such an approach, where agents only need to communicate with nearby agents, was demonstrated in [97].

In [18], a model is developed of the behavior of army ants when selecting traffic lanes. It has been observed that army ants form lanes that maximize traffic flow and in the model, movement rules for individual agents are developed. Each ant in the model avoids collisions and responds to pheromone concentration, and with these simple interaction rules, a sophisticated self-organized structure (collectively-selected, minimally-congested traffic lanes) emerges. In a related work, the behavior of ants are described using a hybrid automaton in [8] and [74] to model the collective-selection of a nest site.

Group size selection was experimentally examined in [50] for shoaling fish. It was reported that the group size of the banded killifish were relatively smaller when individuals thought that food availability was high; subsequently, the group size was relatively larger when individuals thought that the risk of predation was high. An agent-based model of the fish was developed to verify this behavior. The model is an extension of the model provided in [18], where the highest priority for a fish is to avoid obstacles, and in the absence of obstacles, each fish tends to align itself with its neighbors.

A variation of the fish-interaction model of [18] is the model presented in [19], where "informed" agents are included in the network. The effects of such a network on the collective decision-making of the group are presented in [32, 61, 64].

A class of attraction/repulsion functions that achieves swarm aggregation is presented in [30], where inter-agent interactions are based on artificial potential functions. A stability analysis is provided to characterize cohesiveness, size, and motion of the swarm. A Lyapunov approach is used in [57] to identify the conditions under which locally interacting agents can continue to perform cohesive foraging in the presence of noise.

Large-scale biological systems rely on collective decision-making and interaction rules tend to be simple. Moreover, as mentioned in [17], it is unreasonable to assume

that group members in large insect swarms or fish schools have the capacity for individual recognition. For the type of applications we consider in the work, e.g., coordinating a heterogenous network of vehicles, inspiration from such systems falls short. Sophisticated interaction rules are required to assign roles, build preference-based coalitions, and switch between functionalities. To this end, this work focuses on "smarter" animals.

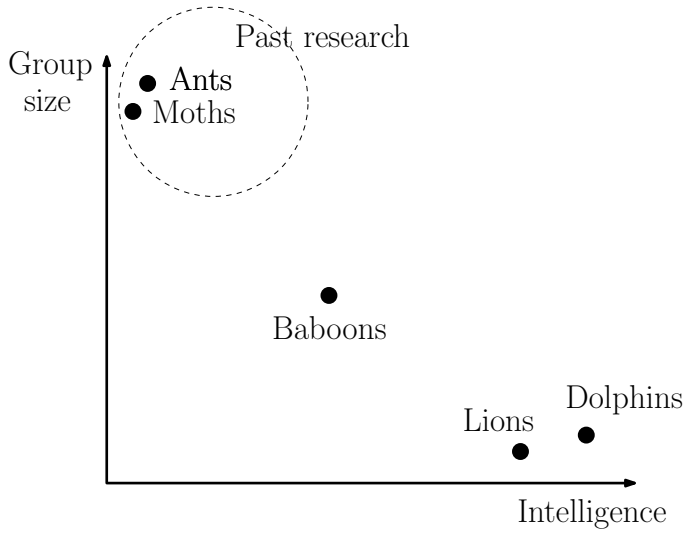


Figure 2: This work mainly focuses on "smarter" animals that tend to live in smaller groups and exhibit sophisticated interaction rules.

Although there is no universal definition for intelligence in animals, one factor that is often used to judge intelligence is the encephalization quotient (EQ) [81]. Humans rank first in the EQ list ($EQ = 7.8$) and animals that have a relatively high EQ tend to live in smaller groups and have complex social interaction rules; examples include Bottlenose dolphins, Savannah baboons, and African lions. The collective choices made in large-scale biological systems arise from reactionary behavior of the individuals, but for more intelligent animals, individuals have preferences over associations with other individuals and these associations are developed over a period of time [17].

The coordination strategies developed in this thesis will primarily focus on Bottlenose dolphins and African lions and detailed observations of their social behaviors

are discussed next. Bottlenose dolphins and African lions are chosen since they exhibit interesting behaviors from a coordination point of view, and for engineered systems, such as a network of autonomous vehicles, these systems will guide us in the design of containment strategies, coalition formation, and team size selection. Both systems utilize highly coordinated foraging techniques to capture prey [23, 75]. Furthermore, male Bottlenose dolphins form multi-level alliances to increase their chances of mating [12] and African lions are known to effectively regulate the size of their aggregations [23].

2.2 Social Behavior of Bottlenose Dolphins

Cetaceans, which include whales and dolphins, are in general intelligent animals. In particular, Bottlenose dolphins, *Tursipos truncatus* rate $2nd$ ($EQ = 5.3$) immediately behind humans in EQ [93], and one example of intelligence is observed when these dolphins wrap marine sponges around their beaks as foraging tools to avoid abrasions [51].

Bottlenose dolphins live in fluid societies, formally known as *fission-fusion societies* [17], where the main group splits up into smaller groups to play, explore, or forage for food, but these groups later rejoin to share food or participate in other activities [58]. These groups may have group leaders, but the entire herd is governed by an overall social hierarchy, where the largest dolphin is usually the most dominant [75]. The most dominant dolphin plays two important roles: 1) it determines the threat level of an environment and 2) it is the first to explore new areas.

The cooperative behaviors that are displayed among dolphins include: advertising resources, foraging and capturing prey, defending group members, and searching for mates by forming alliances [12, 86]. The coordination aspects of foraging and forming alliances are discussed here as they are interesting from a networked controls point of view.

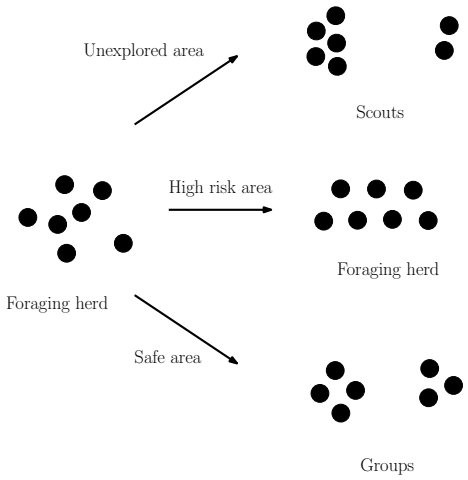
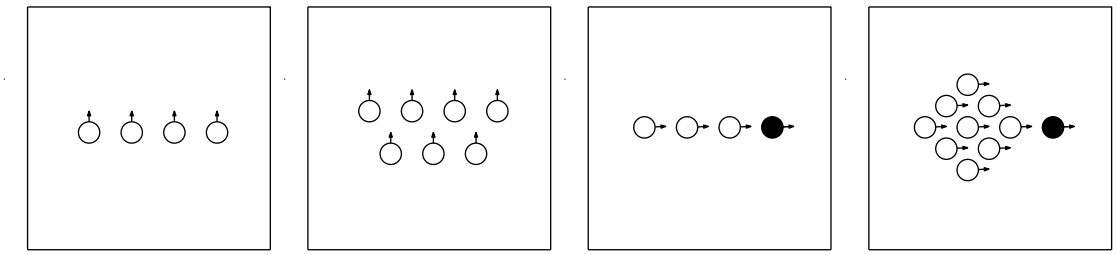


Figure 3: Dolphins forage in three ways - using scouts, as groups, or together as a herd.

The foraging process of Bottlenose dolphins is classified into three phases - *search*, *detect* and *capture* [75]. In the search phase, dolphins look for prey by either sending out scouts (2 dolphins, one of which is the dominant dolphin), by forming groups ($2-6$ dolphins), or by foraging together as a herd (Figure 3). Dolphins in these foraging groups maintain specific formations, which are selected based on the threat level of the environment [75]. The distribution of killer whales and tiger shark (which are dolphin predators), fishing nets, and boating activities constitute of threats [43]. The relative positions of the dolphins in each group are shown in Figure 4. These formations are adaptive since when the dolphins are near shore, where the threat level is high, they tend to stick together and form the *tight* formation, but when foraging further away from the shore, they spread out and switch to either *front* or *double front* formation.

In the detect phase, when a dolphin encounters a sizable amount of prey, i.e. there is enough food available for a group feeding to take place, the rest of the herd will converge to the location of that "advertising" dolphin [75].

In the capture phase, the dolphins have at their disposal a couple of interesting methods to catch prey: *wall method* and *horizontal carousel method*. As mentioned



(a) Front (b) Double-front (c) Line (d) Tight-group

Figure 4: The different types of dolphin formations used during foraging are shown. Some formations require a leader, which is shown in black. Arrows denote headings of the dolphins.

in [75], the success of both methods depends on the ability to constrict the "maneuverability of the prey."

2.2.1 Wall Method

In this method, a group of dolphins drive the school of fish towards a barrier and capture them from the foam of returning water, as illustrated in Figure 5. There are many variations of the wall method (see [75] for details); briefly, in the *fish in front* variation, a group of dolphins use the shore as a barrier. In the *dolphin group as wall* variation, there are two groups of dolphins: one group drives the fish and a second group will act as the wall for the original group. In the *two frontal attacks* variation, there are also two groups of dolphins, but both groups drive the fish towards each other (Figure 6).

2.2.2 Horizontal Carousel

In this method, the dolphins first form a large circle around the school of fish to entrap their prey inside the circle. This carousel-like movement by the dolphins is initiated by either "curving" in from one side of the fish or by simultaneously surrounding them from both sides [75]. Next, the dolphins begin to tighten the encirclement, by forming smaller and smaller circles around the school of fish to constrict the movement of their prey [58], as shown in Figure 7. When the encirclement is small enough, the dolphins

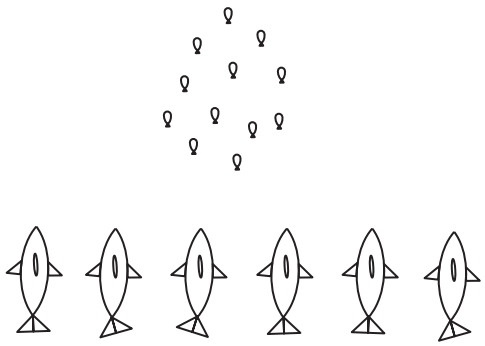


Figure 5: In the fish in front method, dolphins drive fish against the shore.

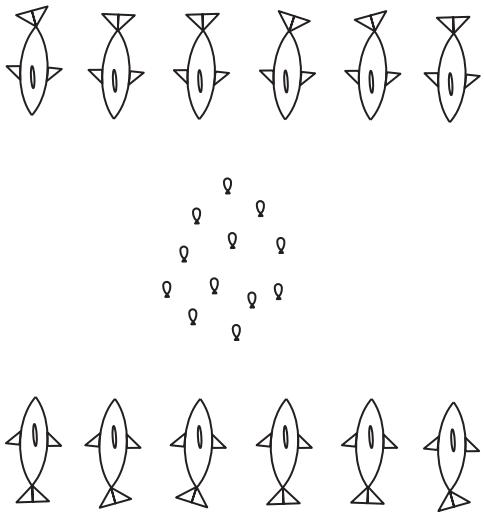


Figure 6: In the two frontal attacks method, groups of dolphins drive fish towards each other.

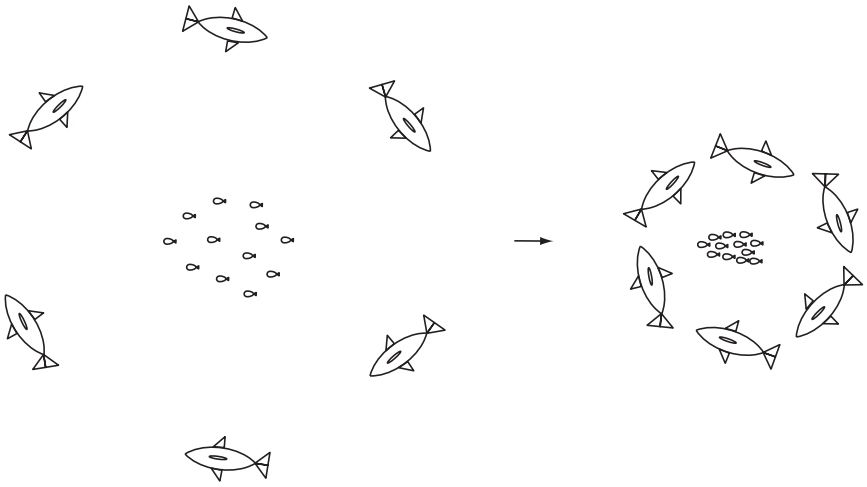


Figure 7: In the horizontal carousel method, dolphins tighten an encirclement around a school of fish.

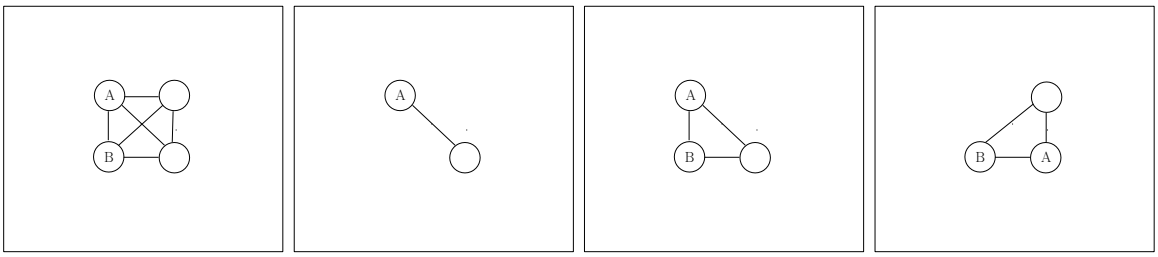
charge into the school and feed on their prey.

The way dolphins charge through the fish is classified into two methods: *kettle method* and *vertical carousel method* [75]. In the kettle method, each dolphin takes a turn to dive into the school of fish, while the others maintain the original encirclement around the prey. In the vertical carousel method, the dolphins feed on the fish by diving into the school all at once. By modeling the foraging techniques of Bottlenose dolphins, we will later show that algorithms can be developed for a group of agents to herd both passive and active objects.

Coordinated behavior of dolphins is exhibited during mating as well. Male Bottlenose dolphins found off the coast of Florida and Western Australia, form varied levels of alliances to capture females and increase their chances of mating [12]. The three levels of alliances are *first-order alliance*, *second-order alliance*, and *super-alliance*.

2.2.3 First-Order Alliance

In this type of an alliance, a pair or a triplet of male dolphins capture a female and herd it by swimming in a specific formation (in the case of a pair, the two males remain slightly behind and on either side of the female) [13].



(a) Sighting 1 (b) Sighting 2 (c) Sighting 3 (d) Sighting 4

Figure 8: Four sightings of dolphins A and B are shown. The two dolphins are seen together in three out of the four sightings.

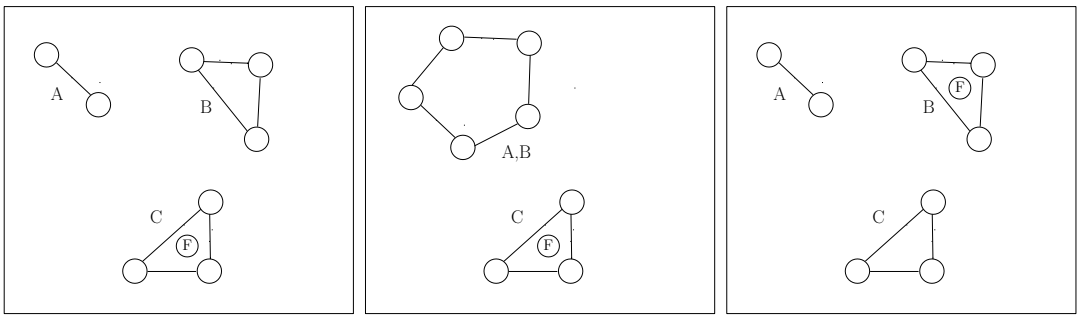
In [13], the authors define an "association coefficient" to identify whether two dolphins are in an alliance and the coefficient is given by

$$\text{association coefficient} = \frac{100 \times 2N_t}{(N_a + N_b)}, \tag{1}$$

where N_t is the total number of party sightings (a party is defined in [13] as a group of dolphins within 10m of each other) in which dolphins A and B are seen together; N_a and N_b are the number of sightings for A and B, respectively. Thus, the association coefficient is an indicator of the degree of cooperation between male dolphins. The coefficient ranges from 0 (two dolphins are never sighted together) to 100 (two dolphins are seen everywhere together). Figure 8 demonstrates this concept, where $N_t = 3$, $N_a = 4$, $N_b = 3$. The association coefficient in this case is 85.7, which is typical for male dolphins in a first-order alliance. This value is in the same coefficient range as those found between females and their nursing calves, and as the calves rarely leave the mother's sight during nursing, this coefficient indicates that a strong bond exists between males in a first-order alliance.

2.2.4 Second-Order Alliance

We use an example to describe the second-order alliance. Consider three first-order alliances: A, B, and C, as shown in Figure 9. As mentioned in [13], it is often the case that a first-order alliance realizes that it cannot steal the female it wants



(a) Alliance C is herding a female.

(b) Alliances A and B form a second-order alliance.

(c) Alliance B steals the female.

Figure 9: An example of a second-order steal.

from another first-order alliance of the same size since the outcome of a fight is unpredictable. Alliance B realizes that to steal the female from Alliance C, it is more favorable to recruit another first-order alliance, which is Alliance A in our example. Alliances between alliances often shift and such multiple levels of alliances, with both hostile and favorable interactions, appear only in dolphins and humans [13].

2.2.5 Super-Alliance

The super-alliance is not as well documented as the two smaller alliances; therefore, this alliance will be excluded from the models produced in Chapter 3. It has been described in [12] as a highly volatile coalition of 14-25 members .

By drawing inspiration from the multi-level alliance forming behavior of Bottlenose dolphins, a model will be developed for a multi-agent system that builds alliances between agents and alliances of alliances. In Chapter 8, this model will be used to provide coordination strategies for a specific engineered application.

2.3 Social Behavior of African Lions

African Lions, *Panthera leo*, live in well-defined social structures known as prides. Typically, these prides consist of $1-3$ adult males and $2-9$ adult females along with their dependent cubs [71]. Males that attach themselves to a group of lionesses, also

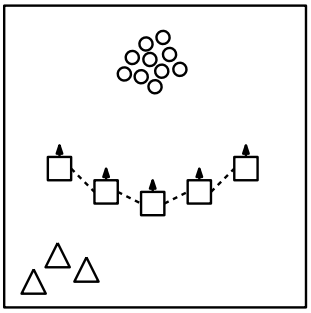


Figure 10: Lionesses (squares) are usually responsible for foraging prey (circles), whereas lions (triangles) are in charge of patrolling the pride area.

known as the resident males, gain significant reproductive advantages over solitary males [9]; as a result, resident males must frequently defend their lionesses from non-resident males [73]. If a group of males successfully take over a pride, by defeating the original resident males, it first ejects the original group of males and then kills their cubs [73]. They breed new cubs with the lionesses to start their own pride; thus, territorial defense is an important task for males to protect their cubs from infanticidal males and in turn, increase genetic fitness [71].

When lionesses have a low success rate of catching a certain prey, they utilize a highly-coordinated group hunting technique where lionesses in the "wing" positions will entrap their prey by driving them towards the lionesses in "center" positions [73]. The resulting shape of the foraging front (Figure 10) is described as a "catcher's mitt" in [23]. Prey caught by females are shared by the entire pride, with males being the first to "claim their share" [23]. Too many females reduce the ability to coordinate and catch prey [85], whereas, too many males result in frequent in-fighting to gain access to females [23].

Bio-inspired rules based on the pride structures of African lions are developed in Chapter 5 to determine the sustainability of group sizes for a network of agents. We consider a group that consists of two classes of agents: one class is responsible for searching an area; the other for providing perimeter security for that area. In this

context, sustainability means the ability of the system to accomplish the task while balancing shared resources.

2.4 Multi-Agent Systems

So far, the discussions on multi-agent systems in this chapter have been biologically motivated. Biology aside, these systems have emerged in numerous engineering applications, e.g., sensor networks, due to benefits like redundancy and cost-effectiveness. In a sensor network, each sensor typically has limited capabilities, making them cost-effective, and a large number of these sensors are deployed so that the failure of some does not affect the completion of a task.

Multi-agent systems consist of decision-making agents, possibly mobile, interacting with other agents over a (dynamic) communication network, and a key challenge in this field is to design coordination rules so that group-level performances are satisfied by locally interacting agents. Next, we will discuss some of the tools available in this field that will be used in our models of biological systems. Also, since the alliance forming and foraging behaviors of dolphins are modeled in the subsequent two chapters, we also discuss previous works involving two multi-agent coordination tasks: coalition formation and the foraging task.

2.4.1 Graph-Based Control

Graph-based control laws for decentralized control strategies have recently received much attention (e.g., [15, 25, 45, 47, 60, 67, 76, 78]). Graph-theoretic methods can describe the network in combinatorial terms and the information available to each agent is modeled through the topology of a graph.

The graph $\mathcal{G} = (\mathcal{V},\ \mathcal{E})$ consists of a vertex set $\mathcal{V}$ and an edge set $\mathcal{E}$. For a multi-agent system with N agents, $\mathcal{V} = \{1, \ldots, N\}$ and $i \in \mathcal{V}$ represents agent i. The edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is used to denote whether a communication link exists between two distinct agents. According to the nearest neighbor-based information exchange

paradigm (see [60]), two agents, i and j, can communicate with each other if they are neighbors. The neighbor set of agent i is denoted by $\mathcal{N}_i$ and $j \in \mathcal{N}_i$ if $(i,\ j) \in \mathcal{E}$.

As mentioned in [21], graph-based control laws abstract away the geometry associated with the communication between agents and the controller only depends on the network topology as opposed to the geometry of the system. Furthermore, analysis tools from graph theory, such as the eigenvalues of the graph Laplacian, can provide useful information about a network [60].

Figure 11 shows three planar agents, each with a communication range of Δ. In this example, $(i,\ j) \in \mathcal{E}(t)$ if $\|x_i(t) - x_j(t)\| \leq \Delta$, where x_i is the position of agent i and $\|\cdot\|$ represents the Euclidean norm (in [15], the resulting graph is referred to as the Δ-disk proximity graph).

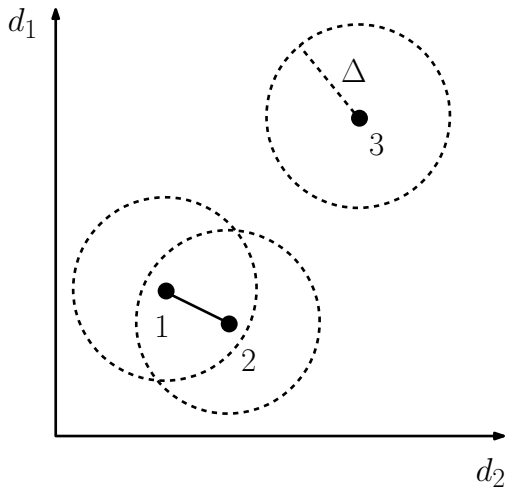


Figure 11: Each agent has a communication range of Δ and exists in a two-dimensional space. Agents 1 and 2 can communicate with each other and form an edge (denoted by the line). Agent 3 cannot communicate with the other two agents.

2.4.2 Consensus Problem

The consensus problem is a canonical problem in decentralized coordination of multi-agent systems [21] and has been studied extensively (e.g., [15], [45], [47], [67]). The goal of the consensus problem is to achieve agreement in the network through a

suitable control strategy. For example, in the rendezvous problem, the goal is to drive all the agents to a common location and a decentralized control strategy that achieves this is

$$\dot{x_i} = -\sum_{j \in \mathcal{N}_i} (x_i - x_j), \tag{2}$$

where x_i is the position of agent i.

The idea of achieving agreement in the network has also been used in the area of formation control. A desired formation can be described in terms of desired inter-agent distances and the goal of each agent is to move in a way that drives the formation error to zero [46].

The widely used nearest neighbor-based interaction rule used for formation control ([25, 47, 67]), consensus problems ([45, 76, 78]), and coverage control ([15, 59]), have a direct biological counterpart [18] and will be an essential component of the biological models produced in the next three chapters.

2.4.3 Coalition Formation

Coalition formation algorithms prescribe rules for agents to form groups in scenarios where a single agent cannot perform a specified task, or the success rate of a single agent performing the task is low. This research topic has received a lot of attention (e.g., see [48, 49, 82, 83, 87, 88, 89, 92]). Most research has focused on creating agent coalitions primarily through two methods, namely, using game theory and social reasoning.

In game theoretic methods, agents are often designed to either maximize their own utilities (selfish agents) or maximize the joint utility of the coalition (unselfish agents) [89]. The social reasoning based algorithms, such as the one in [92], utilizes the ability of an "intelligent agent" to maintain an external description - goals, actions, plans - of other agents in the network. Similarly, other coalition formation algorithms rely on the availability of a "user agents" [62] or "auctioneers" [49]. In [49], the

auctioneer processes requests for proposals (RFPs) issued by agents that desire to be in a coalition. Another approach, as seen in [87], is to find the optimal division of agents through search algorithms.

In Chapter 3, dolphin-inspired coalitions are formed in a decentralized manner without relying on a particular auctioneer-like agent.

2.4.4 Foraging Tasks

The foraging task is known to be a canonical testbed for cooperative robotics [10]. Much of the research in this area, e.g., [5, 26, 52, 55, 90], primarily focused on the search and retrieval of objects scattered in the environment (*source area*) to a target location (*sink area*).

In [90], the effects of physical interference between robots is presented for different foraging strategies. The effects of behavioral diversity of the foraging group is studied in [5], where the behaviors range from "homogeneous" to "specialized." Bio-inspired foraging strategies for static environments, based on ants and bees, are presented in [52] and [55], respectively. In [70], an algorithm is developed that produces a "bucket-brigade-like behavior," where the foraging area is partitioned and each agent is responsible for the partition in which it resides. In [6], the multi-foraging task includes retrieving different types of objects to their designated sinks.

The objects being foraged in both [52] and [70] are passive. However, in [25], the objects are active, and as a result, there are two classes of agents, where one class is herding the other class. The agents are labeled as "leaders" and "followers" and the leader agents use a hybrid control strategy to drive the followers to a target location while ensuring that the followers remain in the convex polytope spanned by the leaders.

CHAPTER III

ALLIANCE FORMING

In this chapter, we model the alliance forming behavior of male Bottlenose dolphins. The goal of this work is to produce a model that is expressive enough to capture multi-level alliances, yet remain simple; and as such, amenable to analysis.

3.1 Introduction

Coalition formation is an important coordination problem in multi-agent robotics when a particular task cannot be accomplished by a single robot. Moreover, there are many situations where the success rate of accomplishing a task by a single coalition is low. For example, consider the surveillance tasks that require teams of UAVs to coordinate with each other over an area of interest [1]. (A more detailed application with a similar setup is the US Navy SEAD mission presented in Chapter 8.) The multi-level alliances of male Bottlenose dolphins were explained in Chapter 2 and based on that behavior, we produce a simple model for a multi-agent system, where 1) agents form coalitions and 2) coalitions form coalitions.

The rest of the chapter is organized as follows: in Section 3.2, we present a hybrid automaton model of first-order alliances. In Section 3.3, we produce a model of first-order alliances based on embedded graph grammars. We model the multi-level alliance forming behavior of dolphins in Section 3.4 and simulations are shown in Section 3.5. Finally, conclusions are provided in Section 3.6 .

3.2 First-Order Alliance Model

In the literature on the social behavior of dolphins, association coefficient represents the camaraderie between two male dolphins. We borrow this terminology and denote

the association coefficient between two agents, i and j, by $\alpha_{i,j}(t)$, at time t. Two new parameters: *familiarity coefficient*, $\phi_{i,j}(t)$, and *rejection coefficient*, $\rho_{i,j}(t)$, are introduced to formulate an agent-based definition of the association coefficient as follows:

$$\alpha_{i,j}(t) = \phi_{i,j}(t) - \rho_{i,j}(t). \tag{3}$$

This formulation is based on the idea that familiarity between dolphins increases their cooperative tendencies, while rejection of forming an alliance by either one increases the animosity between the two [Dr. Lori Marino of the Neuroscience and Behavioral Biology Department at Emory University, personal communication]. A high $\alpha_{i,j}$ indicates that agent i has a strong "desire" to form an alliance with agent j. The coefficient $\rho_{i,j}(t)$ is a measure of the rejection experienced by agent i from agent j, when a request from agent i to form an alliance is denied by agent j, and $\phi_{i,j}(t)$ measures the familiarity that develops between agents i and j. The rejection coefficient evolves according to:

$$\dot{\rho}_{i,j}(t) = \begin{cases} \rho_{max} - \rho_{i,j}(t) & \text{if } j \text{ rejects } i, \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $\rho_{max} > \rho_{i,j}(0) \geq 0$. And, the familiarity coefficient has the following dynamics:

$$\dot{\phi}_{i,j}(t) = \begin{cases} \phi_{max} - \phi_{i,j}(t) & \text{if } j \in \mathcal{N}_i, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where $\phi_{max} > \phi_{i,j}(0) \geq 0$, $\mathcal{N}_i$ denotes the neighborhood set of agent i, and $j \in \mathcal{N}_i$ means that agent j is (geometrically) adjacent to agent i. To insist $\alpha_{i,j}(t) \geq 0 \ \forall \ t$, there is an additional constraint: $\phi_{i,j}(0) \geq \rho_{i,j}(0)$. A bounded association coefficient and first-order differential equations as the dynamics for the rejection and familiarity coefficient are motivated by the goal of creating a simple model, one that captures the first-order alliances of male dolphins and at the same time, remains simple so that it is open to analysis and implementable on engineered systems.

Each agent first identifies, amongst its neighbors, up to two agents towards which it has the highest association coefficients, beyond the threshold $\alpha^\star$ (inspired by the fact that according to the biological definition of (1), association coefficient amongst first-order members is usually above 85). Let $C^1(i)$ be the agent towards which agent i has the greatest association coefficient to form a coalition and in the event of a tie, agent i randomly makes a selection. If there are no ties, $C^1(i)$ is given by

$$C^1(i) = j \mid \alpha_{i,j} > \alpha_{i,m},\ \alpha_{i,j} \geq \alpha^\star\ \forall\ m \in \mathcal{N}_i \setminus \{j\}, \tag{6}$$

and we define the agent $C^2(i)$ as the agent towards which agent i has the second-greatest association coefficient, and this is given by

$$C^2(i) = k \mid \alpha_{i,k} > \alpha_{i,m},\ \alpha_{i,k} \geq \alpha^\star\ \forall\ m \in \mathcal{N}_i \setminus (\{k\} \cup \{C^1(i)\}). \tag{7}$$

For agent i, the two candidates for building an alliance are contained in the *candidate set* $\mathcal{C}(i) = \{C^1(i)\} \cup \{C^2(i)\}$. Since a first-order alliance can contain a maximum of three dolphins, $|\mathcal{C}(i)| \in \{0, 1, 2\}$, where $|\cdot|$ represents cardinality.

Agent i is rejected by agent j if $j \in \mathcal{C}(i)$ and $i \notin \mathcal{C}(j)$. Agents i, j, and k form a first-order pair if they are in each other's candidate sets. Agents i, j, and k form a triplet if each agent is in the candidate set of the other two agents. We will assume that as long as a coalition exists, its members remain neighbors.

3.2.1 Hybrid Automaton Representation

A hybrid automaton is used to model a dynamic system with both continuous and discrete variables, as seen in [44]. Since agents might enter or leave each others candidate sets, the system dynamics will undergo discrete transitions. Hence, we model the first-order alliance as a hybrid automaton, where the continuous dynamics ($\phi_{i,j}(t)$ and $\rho_{i,j}(t)$) unfold within the discrete states. The hybrid automaton for agent i, HA_i is shown in Figure 12. The hybrid automaton for a multi-agent system with N agents is a parallel composition of the automaton of the individual agents, i.e.,

$HA = HA_1 \parallel HA_2 \parallel \ldots \parallel HA_N$. Moreover, Tables 1 and 2 are used to decode the hybrid automaton.

The dynamics and reset conditions of each state is described in Table 1. Other than the initial state, we use the convention `NumberOfCandidates.StatusOfRequest` to name the states in the automaton. For agent i, `NumberOfCandidates`$=|\mathcal{C}(i)|$. If $|\mathcal{C}(i)| = 1$, `StatusOfRequest` $\in \{a, r\}$ and if $|\mathcal{C}(i)| = 2$, then `StatusOfRequest` $\in \{aa, ar, ra, rr\}$, where "a" and "r" represent accept and reject, respectively. Thus, state $2.ar$ indicates that agent i has two candidates, $C^1(i)$ and $C^2(i)$, for forming an alliance and they accept and reject the offer, respectively. The name of a state also represents the set of events that triggers the transition from all other states to that state; it is displayed in bold to indicate it is a set, as shown in Table 2.

Since the automaton is event-driven, there are two possibilities regarding state transitions, namely, synchronized and asynchronized transitions. In the case of synchronized transitions, there is a chance of multiple events occurring simultaneously. To determine the number of events that occur during a transition from state A to state B, we look up the event set $\mathbf{A} \cap \mathbf{B}$ from Table 2. If $\mathbf{A} \cap \mathbf{B} \neq \varnothing$, then only one event causes the transition and if $\mathbf{A} \cap \mathbf{B} = \varnothing$, then two events must occur simultaneously to cause the transition. For example, a transition from state $2.aa$ to $2.rr$ requires two events to fire simultaneously since $\mathbf{2.aa} \cap \mathbf{2.rr} = \varnothing$. This transition is showed in Figure 12(a), which models the case of synchronized transitions. However, for asynchronized transitions, where we assume that the probability of the multiple events firing at the same time is 0, the transition from state $2.aa$ to state $2.rr$ is no longer possible and it is removed from Figure 12(b).

3.2.2 Analysis Results

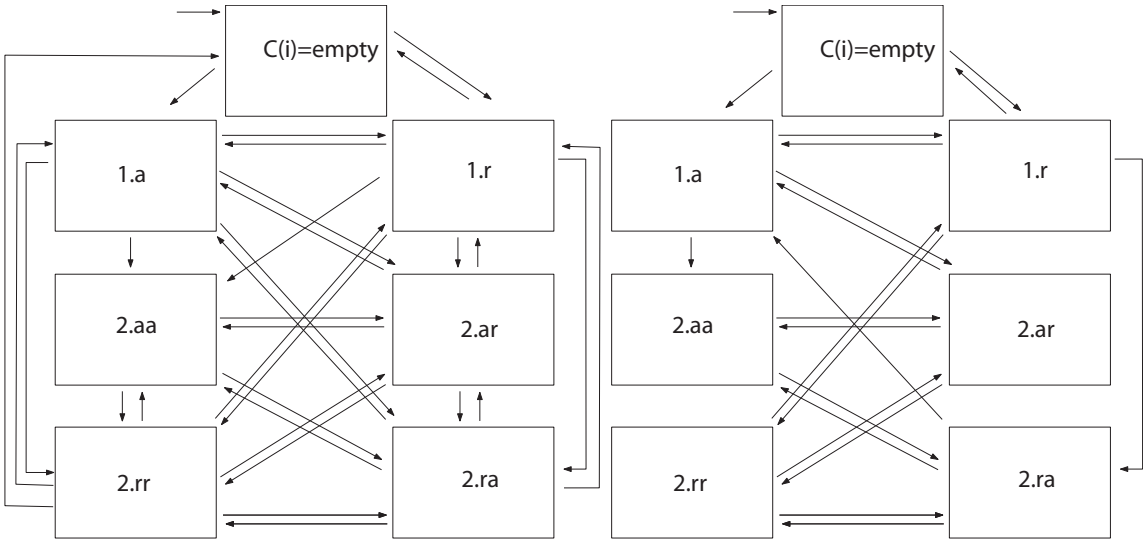
We assume that the candidate set for each agent is updated sequentially, i.e., for agent i, the candidate set is first populated with $C^1(i)$; then by $C^2(i)$. As a result, a

Table 1: Dynamics and reset conditions of the hybrid automata.

State	Dynamics	Reset Condition
$\mathcal{C}(i) = \varnothing$	$\dot{\phi}_{i,j} = \phi_{max} - \phi_{i,j}$ $\dot{\rho}_{i,j} = 0$	$\phi_{i,j}(t) := \phi_{i,j}(0)$ $\rho_{i,j}(t) := \rho_{i,j}(0)\ \forall\ j \in \mathcal{N}_i$
1.a	$\dot{\phi}_{i,j} = \phi_{max} - \phi_{i,j}$ $\dot{\rho}_{i,j} = 0\ \forall\ j \in \mathcal{N}(i)$	$\rho_{i,C^1(i)}(t) := \rho_{i,C^1(i)}(0)$
1.r	$\dot{\phi}_{i,j} = \phi_{max} - \phi_{i,j}\ \forall\ j \in \mathcal{N}_i$ $\dot{\rho}_{i,j} = 0\ \forall\ j \in \mathcal{N}(i) \setminus \{C^1(i)\}$ $\dot{\rho}_{i,C^1(i)} = \rho_{max} - \rho_{i,C^1(i)}$	
2.aa	$\dot{\phi}_{i,j} = \phi_{max} - \phi_{i,j}$ $\dot{\rho}_{i,j} = 0\ \forall\ j \in \mathcal{N}_i$	$\rho_{i,j}(t) := \rho_{i,j}(0)\ \forall\ j \in \mathcal{C}(i)$
2.ar	$\dot{\phi}_{i,j} = \phi_{max} - \phi_{i,j}\ \forall\ j \in \mathcal{N}_i$ $\dot{\rho}_{i,C^1(i)} = 0$ $\rho_{i,C^2(i)} = \rho_{max} - \rho_{i,C^2(i)}$ $\dot{\rho}_{i,j} = 0\ \forall\ j \in \mathcal{N}(i) \setminus \mathcal{C}(i)$	$\rho_{i,C^1(i)}(t) := \rho_{i,C^1(i)}(0)$
2.ra	$\dot{\phi}_{i,j} = \phi_{max} - \phi_{i,j}\ \forall\ j \in \mathcal{N}_i$ $\rho_{i,C^1(i)} = \rho_{max} - \rho_{i,C^1(i)}$ $\dot{\rho}_{i,C^2(i)} = 0$ $\dot{\rho}_{i,j} = 0\ \forall\ j \in \mathcal{N}(i) \setminus \mathcal{C}(i)$	$\rho_{i,C^2(i)}(t) := \rho_{i,C^2(i)}(0)$
2.rr	$\dot{\phi}_{i,j} = \phi_{max} - \phi_{i,j}\ \forall\ j \in \mathcal{N}_i$ $\rho_{i,C^1(i)} = \rho_{max} - \rho_{i,C^1(i)}$ $\rho_{i,C^2(i)} = \rho_{max} - \rho_{i,C^2(i)}$ $\dot{\rho}_{i,j} = 0\ \forall\ j \in \mathcal{N}(i) \setminus \mathcal{C}(i)$	

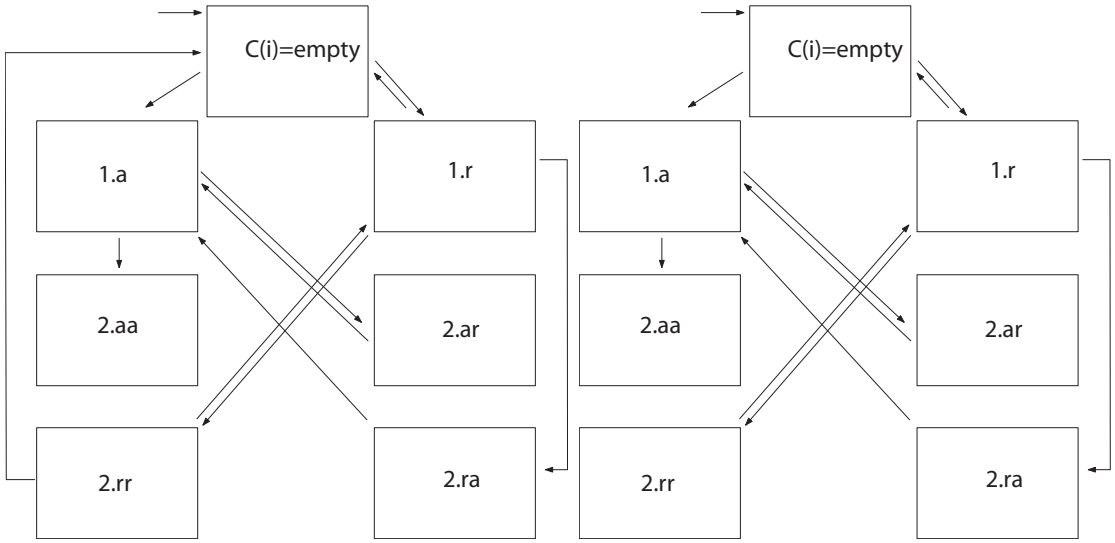
Table 2: Extensional definition of the event sets of the hybrid automata shown in Figure 12. Each element in the set represents an event.

Event Set	Extensional Definition
1.a	$\{C^1(i)\ accepts\}$
1.r	$\{C^1(i)\ rejects\}$
2.aa	$\{C^1(i)\ accepts,\ C^2(i)\ accepts\}$
2.ar	$\{C^1(i)\ accepts,\ C^2(i)\ rejects\}$
2.ra	$\{C^1(i)\ rejects,\ C^2(i)\ accepts\}$
2.rr	$\{C^1(i)\ rejects,\ C^2(i)\ rejects\}$



(a) Synchronous transitions and dynamic neighborhood sets.

(b) Asynchronous transitions and dynamic neighborhood sets.



(c) Synchronous transitions and static neighborhood sets.

(d) Asynchronous transitions and static neighborhood sets.

Figure 12: Hybrid automaton of agent i is shown.

coalition consisting of three agents can form only by adding an agent to an existing pair (for details, see [39]).

Lemma 3.2.1 *State $\mathcal{C}(i) = \varnothing$ can never be reached from the state 1.a in a single transition.*

Proof For agent i in state 1.a, $\phi_{i,C^1(i)}$ monotonically increases and $\rho_{i,C^1(i)} = 0$; as a result, $\alpha_{i,C^1(i)}$ monotonically increases and if $C^1(i) = j$, agent j remains the agent towards which agent i has the greatest association coefficient. Changes in $\mathcal{N}_j$ does not affect $|\mathcal{C}(i)|$ and changes in $\mathcal{N}_i$ leads to two possibilities: either $|\mathcal{C}(i)|$ increases or it remains unaffected. Hence, $|\mathcal{C}(i)|$ cannot decrease in a single transition.

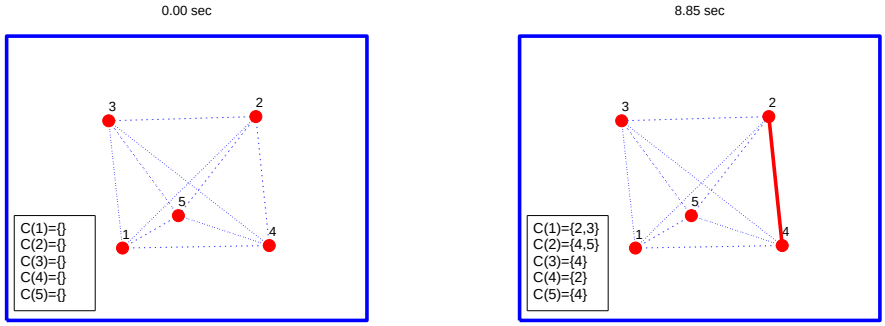
Lemma 3.2.2 *States 1.a and 1.r can never be reached from the state 2.aa in a single transition.*

Proof For agent i in state 2.aa, if $C^1(i) = j$ and $C^2(i) = k$, then changes in either $\mathcal{N}_i$, $\mathcal{N}_j$, or $\mathcal{N}_k$ do not affect $|\mathcal{C}(i)|$. Hence, $|\mathcal{C}(i)|$ cannot decrease in a single transition.

Theorem 3.2.1 *For the case of static neighborhood sets, a first-order alliance (pair or triplet) can never lose alliance members.*

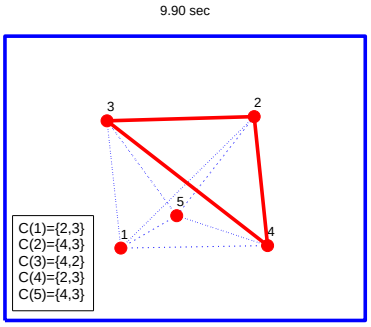
Proof Let agent i be a member of a first-order pair. For static $\mathcal{N}_i$, according to Lemma 3.2.2, from the state 1.a, the possible transitions are to either the state 2.ar or the state 2.aa. In both these new states, the original pair remains intact. In the state 2.aa, the associations among the three members monotonically increase, which explains the deadlock in this state (Figures 12(c)-(d)). From the state 2.ar, agent i can transition only to state 1.a and this occurs when the association towards $C^2(i)$ drops below $\alpha^\star$. Moreover, in this state, the original pair, i and $C^1(i)$, still maintain a coalition.

Next, consider the case where agent i is in a first-order pair but in the state 2.ra. Let $C^1(i) = j$ and $C^2(i) = k$. A transition to the state 1.a occurs once $\alpha_{i,j} < \alpha^\star$ and at this new state, agents i and k remain as a pair, but now $C^1(i) = k$.



(a) Candidate sets are empty.

(b) Agents 2 and 4 form a pair.



(c) Agents 2, 3, and 4 form a triplet.

Figure 13: For static neighborhood sets, the size of a first-order alliance only grows larger. The dotted lines denote edges between agents and the solid line denotes an alliance.

Corollary 3.2.1 *For the case of static neighborhood sets, the size of the first-order alliance can only grow larger.*

3.3 Revised First-Order Alliance Model

Analytical results for the hybrid automaton model of the first-order alliance were derived by restricting the topology of the network, i.e., by focusing on static neighborhood sets. A hybrid automaton can also be used to model the second-order alliance, where alliances are formed at two levels - between agents to form first-order pairs or triplets, and between alliances to steal/defend females. In fact, a nested hybrid-automaton may be needed, where at the lower level, the automaton described in the

previous section builds first-order alliances, whereas at the top level, an automaton can be designed to form alliances between alliances.

However, such an approach, with the addition of interaction rules between female agents and male agents (either solitary or in a coalition), can become cumbersome. Recall that our goal is to capture the biological behavior with simple agent-based models. To this end, we use *embedded graph grammars* (EGGs) to describe the multi-level alliance model from a higher level of abstraction. The lower level rules, e.g., where agents form first-order alliances, are still handled using hybrid automata models.

To model the multi-level alliances of dolphins, we need a heterogeneous network that consists of both male and female agents. Let $\mathcal{V}^m = \{1, \ldots, N^m\}$ denote the set of male agents in the network and similarly, let $\mathcal{V}^f = \{N^m + 1, \ldots, N^m + N^f\}$ denote the set of female agents.

An EGG is a formalism that takes a vertex-labeled graph as an input and yields another vertex-labeled graph as an output, based on a rule set (see [95] for a formal definition). In our model, the vertex-labeled graph will be defined as $\mathcal{G} = (\mathcal{V},\ \mathcal{E},\ l,\ \Sigma)$, where $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges. We assume that each agent has a communication range of Δ and two distinct agents, i and j, form an edge, i.e., $(i,\ j) \in \mathcal{E}$ if $\|x_i - x_j\| \leq \Delta$, where x_i is the position of agent i and $\|\cdot\|$ represents the Euclidean norm. The communication between agents is modeled using the nearest-neighbor rule, i.e., agents i and j can communicate if they are neighbors; thus, $j \in \mathcal{N}_i$ if $(i,\ j) \in \mathcal{E}$. The function l assigns a label from the label set, Σ, to the agents in the network, $\mathcal{V} = \mathcal{V}^m \cup \mathcal{V}^f$.

Associated with G is a set of transition rules, Φ. Each rule in the set Φ is given by a pair $r = (L \rightharpoonup R)$, where L and R are subgraphs. Furthermore, there is a guard associated with each rule r that evaluates to either `true` or `false`. If the guard condition associated with rule r is `true`, then the application of the rule r on the original subgraph L produces the subgraph, R. In fact, the guard conditions are

state-dependent since the evaluation of a guard condition depends on the geometric adjacency of the nodes in the subgraph.

The set of all solitary dolphins, first-order pairs, and first-order triplets is denoted by $\mathcal{D}$, $\mathcal{P}$, and $\mathcal{T}$, respectively, where $\mathcal{D} \subseteq \mathcal{V}^m$, $\mathcal{P} \subset \mathcal{V}^m \times \mathcal{V}^m$, and $\mathcal{T} \subset \mathcal{V}^m \times \mathcal{V}^m \times \mathcal{V}^m$. For first-order alliances, the label set is defined as $\Sigma^1 = \{w,\ p,\ t\}$, where the labels w, p, and t represent the modes *wander*, *pair*, and *triplet*, respectively. Building first-order alliances is represented by the following grammar transition rules:

$$\Phi^1 = \begin{cases} w \quad w \ \rightharpoonup \ p \text{—} p & (r_1), \\ \begin{array}{c} p \\ / \\ p \end{array} \quad w \ \rightharpoonup \ \begin{array}{c} t \\ /\ \backslash \\ t \text{—} t \end{array} & (r_2), \end{cases} \tag{8}$$

where $l(i) = w, \forall\ i \in \mathcal{D}$. For two wandering agents i and j, i.e., $l(i) = w$ and $l(j) = w$, the guard condition for r_1 is `true` if $i \in \mathcal{C}(j)$ and $j \in \mathcal{C}(i)$. The outcome of applying r_1 is $(i,\ j) \in \mathcal{P}$, where $l(i) = p$ and $l(j) = p$. Similarly, for a wanderer and a first-order pair, i.e., $l(i) = p$, $l(j) = p$, and $l(k) = w$, such that $(i,\ j) \in \mathcal{P}$, the guard condition for r_2 is `true` if each agent is in the candidate set of the other two agents. The outcome of applying rule r_2 is $(i,\ j,\ k) \in \mathcal{T}$, where $l(i) = t$, $l(j) = t$, and $l(k) = t$. Notice that the guard conditions for r_1 and r_2 are evaluated using the method prescribed in Section 3.2.

The breakdown of a first-order alliance can be described by the following transition rules:

$$\Phi^2 = \begin{cases} p \text{—} p \ \rightharpoonup \ w \quad w & (r_3), \\ \begin{array}{c} t \\ /\ \backslash \\ t \text{—} t \end{array} \ \rightharpoonup \ \begin{array}{c} p \\ / \\ p \end{array} \quad w & (r_4), \\ \begin{array}{c} t \\ /\ \backslash \\ t \text{—} t \end{array} \ \rightharpoonup \ w \quad w & (r_5). \end{cases} \tag{9}$$

A first-order pair breaks down, i.e., the guard condition for r_3 is `true`, if both agents are no longer simultaneously in each other's candidate sets. The guard condition for

r_4 is `true` if there a single pair-wise rejection between the agents in the triplet. The guard condition for r_5 is `true`, if there are two pair-wise rejections between the agents in the triplet.

Before describing more transition rules, it is important to describe the implementation of EGG rules. The implementation is based on an *extended token based* system described in [95]. In short, the protocol selects an agent from a randomly-generated sequence and gives that agent sole control to apply EGG rules and update its own labels and the labels of agents involved in the application of that rule. These agents are then placed in a *used set* to denote the fact that they are unavailable for further relabeling when the next agent chosen from the sequence is in charge. For example, consider a network with three agents, $\{1,\ 2,\ 3\}$, with the token $\{3,\ 2,\ 1\}$. Agent 3 applies a rule and if this involves agent 2, then $used = \{3,\ 2\}$. It is now agent 1's turn but since it cannot include agents from the used set, no more rules can be applied and it places itself in the used set. Once a sequence is exhausted, a new sequence is generated and the used set is cleared. This protocol also avoids rules being implemented concurrently and ensures that the graph information is not modified without applying the appropriate rules.

3.3.1 Modes of a First-Order Alliance

The goal of a first-order pair is to capture females, but this is only true when the dolphins in the alliance desire to mate. We characterize this desire to herd a female by the herding function, $\eta : \{\mathcal{P}, \mathcal{T}\} \times \mathbb{R} \to \mathbb{R}$. When a pair or a triplet is first formed, its herding coefficient is given by $\eta(i,\ t) = 0,\ \forall\ i \in \{\mathcal{P}, \mathcal{T}\}$. Thresholds $\eta_1^\star$ and $\eta_2^\star$ are used to classify the first-order alliance into three modes according to the hybrid automaton of Figure 14, where $\eta_2^\star > \eta_1^\star$. When an alliance is formed, it is initially in the *search* mode and when it herds a female it enters the *herd* mode, where the herding function begins to increase. For herding alliance i, if $\eta(i,\ t) \geq \eta_1^\star$, it releases

the female and enter the *idle* mode, where its herding function continues to increase. When $\eta(i,\ t) \geq \eta_2^\star$, the idle alliance i transitions to the *search* mode.

We assume that the membership of an alliance cannot be modified (no new members are recruited and no members leave) when an alliance is herding a female. Thus, in the herd mode, the guard conditions of rules r_1 and r_2 always evaluate to `false`.

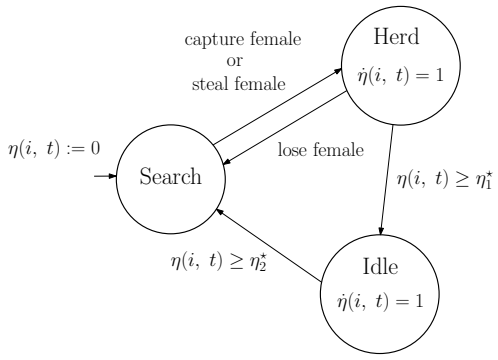


Figure 14: The hybrid automaton classifying the modes of a first-order alliance i, where $\eta_2^\star > \eta_1^\star$.

3.3.2 Male-Female Interactions

The female agents in the network can either be in the *free* mode or in the *captured* mode; as a result, the label set for female agents is given by $\Sigma^f = \{f,\ f'\}$, where a female labeled "f" is not being herded by any alliance, while a female labeled "f'" is being herded by a first-order alliance. The set of all free females is given by $\mathcal{F} = \{i \in \mathcal{V}^f \mid l(i) = f\}$ and the set of all captured females is given by $\mathcal{F}' = \{i \in \mathcal{V}^f \mid l(i) = f'\}$.

The male-female interactions are characterized using the following assumptions:

1. A single male agent cannot herd a female,

2. A first-order alliance can always capture a free female, and,

3. A first-order alliance can herd exactly one female at a time.

Since *all* the agents in the network have a limited communication range, when an

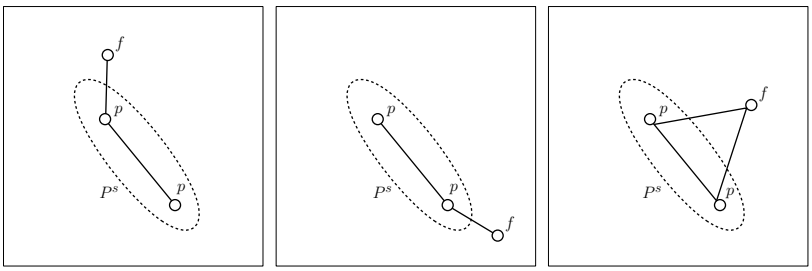


Figure 15: The three ways in which a searching pair can detect a female are shown. In the transition rules, the two agents labeled p are replaced by a "supernode" P^s for simplicity of notation.

agent interacts with an alliance, there are a variety of ways in which this interaction can take place. For example, consider a free female agent interacting with a searching pair. In this case, the female could be communicating with either a single member of the pair, or with the entire pair at once (Figure 15). In any case, the outcome is the same: the free female will be captured by the pair. Thus, instead of explicitly describing transition rules for each possible scenario regarding interactions with alliances, a shorthand notation is used (see [35] for details). The label set $\Sigma^2 = \{P^s,\ T^s,\ P^h,\ T^h,\ P^i,\ T^i\}$ is used to classify the modes of a first-order alliance. The superscripts "s", "h", "i" represent the search, herd, and idle mode, respectively.

Based on this simplified notation for alliances, we have the following grammar transition rules regarding males in a first-order alliance:

$$\Phi^3 = \begin{cases} F^s \quad f \quad \rightharpoonup \quad F^h - f' & (r_6), \\ F^h - f' \quad \rightharpoonup \quad F^i \quad f & (r_7), \\ \qquad F^i \quad \rightharpoonup \quad F^s & (r_8), \end{cases} \tag{10}$$

where, $F \in \mathcal{P} \cup \mathcal{T}$. The guard condition for r_6 is `true` as long any agent from the alliance can detect the free female. The guard conditions for r_7 and r_8 are evaluated using the automaton of Figure 14. For a first-order alliance i, the rule r_7 is applied if $\eta(i,\ t) \geq \eta_1^\star$, and the rule r_8 is applied if $\eta(i,\ t) \geq \eta_2^\star$.

According to [13], alliances prefer some females to others. We define a function

$\gamma : \{\mathcal{P},\ \mathcal{T}\} \rightarrow \mathcal{V}^f$ to identify the female being pursued by an alliance. An alliance can detect multiple females in a variety of different situations. Let us consider a pair $(i, j) \in \mathcal{P}$ and two distinct females a and b. In one scenario, agent i detects both a and b. In this case, according to our model, agent i will prefer a free female over a captured one, but if both females have the same label, agent i makes a random selection. If agent i select a, then $\gamma((i, j)) = a$. In the other scenario, agent i and j could each detect a separate female, e.g, agent i detects a and agent j detects b. The extended token based protocol resolves this situation, e.g., if it is agent i's turn to apply EGG rules, it will select female a, and set $\gamma((i, j)) = a$.

The interactions between a first-order alliance and a captured female (the premise for creating second-order alliances) is the topic of the next section.

3.4 Second-Order Alliance Model

When a first-order alliance is attracted to a female that is already being herded, it tries to recruit another alliance in order to defeat the herding alliance and steal the female. In some cases, an alliance will actually help the herding alliance defend the female against other alliances [13]. In our model of the second-order alliance, alliances will recruit alliances based on factors such as conflict of interest and performance index. An alliance's ability to steal or defend a female will be based on the outcome of a "fight."

3.4.1 Recruiting an Alliance

A first-order alliance that is pursuing a captured female or a first-order alliance that notices that another alliance is pursuing the female it is herding, will both try to recruit other first-order alliances to form a second-order alliance. Although both alliances will initiate a second-order alliance, they will do so for different reasons, the former for offensive purposes and the latter for defensive purposes.

We denote the set of all first-order alliances that are in the search mode, the

herd mode, and the idle mode, by the sets $\mathcal{S}$, $\mathcal{H}$, and $\mathcal{I}$. The alliances in the search mode will initiate a second-order alliance by recruiting alliances that are either in the herd or idle mode since two alliances in the search mode have a conflict of interest. Similarly, two alliances in the herd mode cannot form a second-order alliance. We will assume that the only alliances available to respond to the request of forming a second-order alliance are the alliances in the idle mode.

We denote $\mathcal{U}$ as the set of alliances that initiate a second-order alliance. Consider an alliance i pursuing a female a, i.e., $\gamma(i) = a$. If $\gamma(i) \in \mathcal{F}'$, then alliance i will initiate a second-order alliance, $i \in \mathcal{U}$, to steal a. In another scenario, consider an alliance i that is herding a female a. If $n \in \mathcal{N}_m$, such that m is in alliance i and n is in alliance j, where $\gamma(j) = a$, then alliance i will initiate a second-order, $i \in \mathcal{U}$, to defend a.

The alliance that initiates a second-order, must recruit an alliance in the idle mode without losing its communication link with the female it is either defending or trying to steal. In our model, the searching alliance circles the female it wants to steal with a radius of Δ (the communication range of an agent) until it is able to recruit an idle alliance. This female may be moving since the alliance herding it might be moving itself to recruit an idle alliance.

In our model, when an alliance has a choice between joining two first-order alliances, it will make its decision based on a performance index, defined as $\pi : \{\mathcal{P},\ \mathcal{T}\} \times \{\mathcal{P},\ \mathcal{T}\} \to \mathbb{R}$, where $\pi(i, j)$ is the performance index between first-order alliances i and j. When given a choice between two first-order alliances, an alliance will select the alliance with which it shares the greater performance index. The formation of a second-order alliance is described using the following grammar transition

rules:

$$\Phi^4 = \begin{cases} F^s \quad F^i \rightharpoonup F^s \text{——} F^i \quad (r_9), \\ \quad f' \qquad\quad\ f' \\ \quad\diagup \rightharpoonup \quad\diagup \\ F^h \quad F^i \quad F^h \text{——} F^i \quad (r_{10}), \end{cases} \tag{11}$$

where $F \in \{P \in \mathcal{U},\ T \in \mathcal{U}\}$. According to the token, if its agent m's turn to apply EGG rules, where m is in alliance $i \in \mathcal{U}$, then it will send a request to alliance $j \in \mathcal{I}$ if $\pi_{i,j} > \pi_{i,k}$, $\forall\ n, p \in \mathcal{N}_m$, where n is in alliance j and p is in alliance $k \in \mathcal{I}$. If there is a tie, agent m will send a request to an idle alliance based on a random selection. We define the function $req : \mathcal{U} \to \mathcal{I}$ to identify which idle alliance is being sent a request.

Similarly, if its agent m's turn to apply EGG rules, where m is in an alliance $i \in \mathcal{I}$, $req(j) = i$, and $req(k) = i$, alliance i will accept alliance j's request if $\pi_{i,j} > \pi_{i,k}$, $\forall\ n, p \in \mathcal{N}_m$, n is in alliance j and p is in alliance k. Again, ties are resolved based on a random selection. The update rule for the performance index depends on the outcome of a fight and that is discussed in the next section.

3.4.2 Outcome of Fights

The second-order alliance between dolphins only last during the interval of a fight [14]. In our model, we do not explicitly model a fight, rather, alliances are constantly competing to gain access to female agents, and an agent initiating a second-order can be thought of as the start of a fight. We model the outcome of a fight by either a female being stolen or being successfully defended. The alliance that steals the female after initiating the second-order wins the fight; similarly, an alliance that successfully defends its female after initiating a second-order wins a fight. Winning a

fight improves the performance index between the alliances of the successful second-order as follows:

$$\dot{\pi}_{i,j}(t) = \begin{cases} \pi_{max} - \pi_{i,j}(t) & \text{if } i \text{ and } j \text{ win a fight,} \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

where $\pi_{i,j}(0) = 0$ and $\pi_{i,j} = \pi_{j,i}$ for any two first-order alliances, i and j. In our model, the outcome of a fight is decided by size. The larger alliance wins the fight and the following transition rules describes this situation:

$$\Phi^5 = \begin{cases} F^s - F^i \quad \tilde{F}^h - f' \rightharpoonup F^h - f' \quad F^i \quad \tilde{F}^h \quad (r_{11}), \\ \begin{array}{ccc} F^s & f' & \\ \diagup & \diagup & \\ F^i & \tilde{F}^h & \text{---} \ \tilde{F}^i \end{array} \rightharpoonup \begin{array}{ccc} & f' & \tilde{F}^s \\ \diagup & & \tilde{F}^i \\ F^h & F^i & \end{array} \quad (r_{12}), \\ \begin{array}{ccc} F^s & f' & \\ \diagup & \diagup & \\ F^i & \tilde{F}^h & \text{---} \ \tilde{F}^i \end{array} \rightharpoonup \begin{array}{ccc} & f' & F^s \\ \diagup & & F^i \\ \tilde{F}^h & \tilde{F}^i & \end{array} \quad (r_{13}), \end{cases} \tag{13}$$

where $F \in P,\ T$. With the applications of rules r_{11}-r_{13}, no new labels are produced in the right graph; instead, some alliances may interchange modes. To keep track of the alliances in the left graph that switch modes in the right graph, we use the notation $\tilde{F}$ along with F. Since the outcome of a fight is determined by the size of the second-order alliances, the guard condition for rule r_{11} is always `true`. Moreover, the guard condition for rule r_{12} is `true` if the second-order alliance attempting a steal is larger than the defending second-order alliance and the guard condition for rule r_{13} is `true` if the defending second-order alliance is either larger or of the same size as the second-order alliance attempting the steal.

The label and rule sets for the entire network is given by $\Sigma = \Sigma^1 \cup \Sigma^2 \cup \Sigma^2 \cup \Sigma^f$ and $\Phi = \Phi^1 \cup \Phi^2 \cup \Phi^3 \cup \Phi^4 \cup \Phi^5$, respectively. Based on this model of building second-order alliances, we can now present analytical results.

Theorem 3.4.1 *In a network consisting of three first-order alliances in three different modes and a single captured female, if the alliance that is herding the female loses it to a steal, then it cannot steal back the female, without anything else changing in the network.*

Proof Let the alliances in the network be denoted by $i \in \mathcal{S}$, $j \in \mathcal{H}$, and $k \in \mathcal{I}$. Alliance j will lose its female if and only if $\pi_{i,\,k} > \pi_{j,\,k}$. After the fight, $i \in \mathcal{H}$, $j \in \mathcal{S}$, and $k \in \mathcal{I}$. Also, the outcome of the fight further increases $\pi_{i,\,k}$, while $\pi_{j,\,k}$ remains unchanged. Thus, as long as $i \in \mathcal{H}$, $j \in \mathcal{S}$, and $k \in \mathcal{I}$, an attempt by alliance j to steal back the female will be unsuccessful.

Corollary 3.4.1 *In a network consisting of three first-order alliances in three different modes and a single captured female, the only way to recapture a lost female is to form a new second-order alliance.*

From Corollary 3.4.1, we observe that the results of a lost fight cannot be overturned if the losing first-order alliances stay in the same second-order alliance.

3.5 Simulations

Here we model the agents as unicycles with constant speed and random change of headings w_i and let them roam a confined area, $\mathcal{A}$, as follows:

$$\begin{aligned} \dot{x}_i &= V\cos(\theta_i), \\ \dot{y}_i &= V\sin(\theta_i), \\ \dot{\theta}_i &= w_i. \end{aligned}$$

The agents bounce off the virtual walls when they hit the boundary of the predefined space. We pre-specify a constant radius of interaction, Δ, and randomly initialize familiarity and rejection coefficients. Members of a coalition will follow the agent with the lowest index in the coalition. In applying the EGG rules, each agent

takes its turn with the priority given to the agent with the lower index. This ensures proper application of EGGs as no two agents cannot simultaneously update the same information in the network.

Figure 3.5(a) depicts the formation of first-order coalitions (labeled P and T) through interacting wanderer agents w. Upon formation, first-order alliances enter the search mode (indicated by the subscript s).

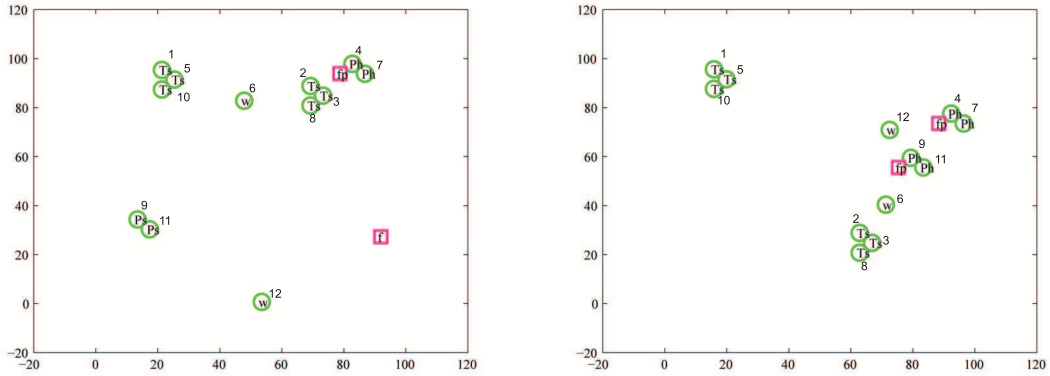
In Figure 3.5(a), $\mathcal{T} = \{(1, 5, 10), (2, 3, 8)\}$ and $\mathcal{P} = \{(4, 7), (9, 11)\}$ are formed. Coalition $(4, 7)$ is herding a female and its label changes from P to Ph. This herded female is within the sensing range of coalition $(2, 3, 8)$; hence, coalition $(2, 3, 8)$ begins to look for an idle first-order coalitions to steal this female. Noticing this, coalition $(4, 7)$ also initiates a second order alliance to defend its female.

Figure 3.5(b) illustrates the state of this system after some time, coalition $(2, 3, 8)$ is unsuccessful in its steal as no idle first-order coalitions could be found, and pair $(9, 11)$ is herding the second female.

Figure 3.5(c) depicts the release of the female by $(4, 7)$ since coalitions remain in the herding mode for a pre-specified time. It subsequently enters the idle mode and is labeled Pi. Meanwhile, coalition $(1, 5, 10)$ has lost one of its members since in our model, first-order coalitions in the search mode can break down. The resulting pair, $(1, 5)$, is interested in the female herded by $(9, 11)$. It begins circling the female and looking for an idle pair to steal the female.

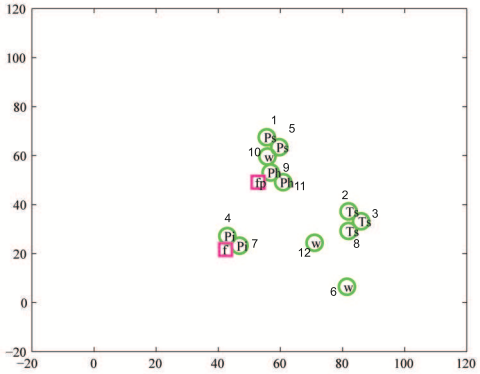
3.6 Conclusions

The main result of this chapter states that in a network with three first-order alliances representing three different modes, search, herd, and idle, if the herding alliances loses its female to a steal, it cannot re-capture its female without anything else changing in the network. This result has been observed in the field in numerous occasions



(a) Wanderers, pairs, and triplets are labeled P, T, and w, respectively.

(b) $(9, 11)$ switches from search to herd mode.



(c) Second-order $(9, 11, 1, 5, 10)$ is initiated by $(9, 11)$ in response to $(2, 3, 8)$'s threat.

Figure 16: Multi-level coalition formation.

[Dr. Richard Connor, Department of Biology at University of Massachusetts, personal communication], where alliances immediately attempt to steal the female lost to another alliance. These attempts are unsuccessful and this shows that our simple agent-based model is expressive enough to mimic aspects of alliance interactions that are observed in nature.

CHAPTER IV

FORAGING

This chapter presents a model of the foraging behavior of Bottlenose dolphins. Dolphins are modeled as first-order systems in which interactions are defined through spatial proximity. A hybrid automaton is used to describe the entire foraging process - search, detect, and capture - and simulation results illustrate the richness of the model. Also, three formation selection strategies are presented based on the horizontal carousel method to capture fish.

4.1 Introduction

The foraging process of Bottlenose dolphins was discussed in Chapter 2. Here, using tools from decentralized networked controls and hybrid systems, a model of this behavior is developed in the context of multi-agent herding. In [26], a "stop/go" policy was developed to confine the herded agents in the convex polytope of the herding agents under the assumption that the herded agents were *cooperative*. Due to the predatory nature of dolphins towards fish, with the model developed here, we simulate the confinement of non-cooperative agents. The goal is to produce a model that is rich enough to capture the foraging behavior, yet lends itself to further analysis. Both classes of agents (dolphins and fish) are modeled as first-order networks with unicycle dynamics, where decisions by these autonomous agents are made solely based on local interactions with other agents.

The rest of the chapter is organized as follows: in Section 4.2 a hybrid automaton is developed to model of the foraging behavior of dolphins. Section 4.3 presents the predator-prey interaction model and simulation results are shown in Section 4.4. Decentralized strategies to select formations are discussed in Section 4.5 and conclusions

are provided in Section 4.6.

4.2 Multi-Agent Herding

The result of this section is the development of a hybrid automaton that models the selection of a suitable search method and prey-capture technique based on factors like the availability of prey and the threat level of the environment.

4.2.1 Search Method Selection by Leader Agents

As mentioned in Chapter 2, dolphins have a well defined social hierarchy, where the role of the dominant dolphin goes to the largest male. The role of this dolphin is explained in [75] in much detail, but to summarize, the most dominant dolphin in the herd plays two important roles: 1) it determines the threat level of an environment and 2) it is the first to check out an unexplored area.

We establish this notion of dominance for our multi-agent system by assigning each agent $i \in \mathcal{H}$ (we use $\mathcal{H}$ to denote "herd") a dominance factor $d_i \in \mathbb{R}$. We assume that there exists an agent $l \in \mathcal{H}$ such that $d_l > d_i,\ \forall\ l \neq i \in \mathcal{H}$. With this formulation, we establish the presence of a "leader" agent within the entire set of agents.

Since the herd can split into groups, we also assume every group within the herd is also led by a dominant dolphin from within that group. Thus, there is a dominant dolphin that leads the entire herd, but when a task requires them to cluster into groups, each group is also led by a dominant dolphin from within the group.

If there are N_G groups ($N_G = 1$ when the entire herd forages together) that emerge from the herd, we have $\mathcal{H} = \cup_{j=1}^{N_G} \mathcal{G}_j$ such that $\mathcal{G}_j \neq \varnothing$ and $\mathcal{G}_j \cap \mathcal{G}_i = \varnothing,\ \forall\ j \neq i \in \{1, \ldots, N_G\}$.

As mentioned earlier, we assume that each group also has a leader, i.e., there exists an agent $l_j \in \mathcal{G}_j$ such that $d_{l_j} > d_i,\ \forall\ l_j \neq i \in \mathcal{G}_j$. The set of leaders, $\mathcal{L}$, is given by $\mathcal{L} = \{l_j \mid l_j \in \mathcal{G}_j\ \forall\ j \in \{1, \ldots, N_G\}\}$. Without loss of generality we assume

the most dominant agent in the entire herd belongs to the first group and denote it as l_1.

Since the dominant dolphin is always monitoring the threat level of the environment, we define a function $p^r : \mathcal{L} \to \mathbb{R}$ to characterize the threat assessment ability of leader agents. Thus, $p^r(i)$ denotes the threat level of the environment assessed by agent $i \in \mathcal{L}$ (superscript "r" stands for risk). The critical threat levels r_{min} and r_{max} will be used to classify predator risk into different levels, where $0 < r_{min} < r_{max} \in \mathbb{R}$. In our model, we allow group leaders to monitor the threat level, in addition to the leader agent l_1, to avoid the scenario where the agent l_1 declares an area to be safe while a group leader further away is being encountered by a predator.

If the dolphins are foraging in a previously unexplored area, then the dominant dolphin will form a group of "scouts" with other lesser-dominant dolphins in the herd to examine the area while searching for food, while the main herd in turn, follows these scouts from a safe distance [16]. We characterize the familiarity of a foraging area through the function $area : l_1 \to \{0, 1\}$ such that $area(l_1) = 1$ denotes that the current foraging area has previously been used to forage, while $area(l_1) = 0$ represents a previously unexplored foraging area. Notice that this assessment is done by the most dominant agent in the herd, l_1, and when $area(l_1) = 0$, the dolphins will search for food using scouts; otherwise, the dolphins will either forage in groups or as a whole herd. At the start of the foraging process, a suitable search method is chosen by the leader agent l_1, depending on the risk of predation and the familiarity of the foraging area using the following criteria:

$$\text{search} = \begin{cases} \text{scouts} & \text{if} \quad area(l_1) = 0 \wedge p^r_{max} < r_{max} \\ \text{group} & \text{if} \quad area(l_1) = 1 \wedge p^r_{max} < r_{min} \\ \text{herd} & \text{if} \quad area(l_1) = 1 \wedge r_{min} \leq p^r_{max} < r_{max} \end{cases}$$

where $p^r_{max} = \max_i p^r(l_i)$. Notice that if the threat level is higher than r_{max} before foraging begins, then the leader agent prevents the entire herd from starting the

process.

4.2.2 Communication Topology

Dolphins use their echolocation system to communicate and this constitutes of using sonar and making "rapid clicks" (see [16, 58]). We model this limited range communication using an undirected edge set (as seen in Chapter 3); as such, $(i, j) \in \mathcal{E}(t)$ if $\|x_i(t) - x_j(t)\| \leq \Delta$, where $x_i(t)$ is the position of agent i at time t and Δ is the communication range. The neighborhood set of agent i is denoted by $\mathcal{N}_i$, and $j \in \mathcal{N}_i$ if $(i, j) \in \mathcal{E}(t)$. We further assume that group leaders always remain connected, i.e., $(l_i, l_j) \in \mathcal{E}(t),\ \forall\ l_i, l_j \in \mathcal{L}$.

We postulate the nearest-neighbor rule to model both the inter-dolphin and inter-fish interactions. In [86], it is observed that individual dolphins in a group move with a virtual bubble around them that other members do not enter. Moreover, small fish generally maintain a constant "inter-individual distance" as described in [50] and from a networked control point of view, this implies that each agent is aware of the position of its neighbors. As a result, all agents in our model are capable of computing their relative displacements from their neighbors.

4.2.3 Detection

When a dolphin encounters a school of fish, if there is enough prey available for a group feeding to take place, the rest of the herd will come and join this "advertising" dolphin [75]. This implies that the location of the prey is sent to the other dolphins and that dolphins are capable of estimating the biomass of the prey they encounter. We instill this measuring capability in our agents through the function $p^f : \mathcal{H} \rightarrow \mathbb{R}$. Thus, $p^f(i)$ represents agent i's estimate of the amount of prey it encounters (superscript "f" stands for fish). If this estimate is greater than the threshold, $f^\star \in \mathbb{R}$, then the agent will "advertise" the location of the prey to the rest of the herd. In our formulation, the location being broadcasted is the centroid of the prey, ρ. The advertising aspect

is recreated by the function $adv : \mathcal{H} \rightarrow \{0, 1\}$, defined as follows:

$$adv(i) = \begin{cases} 1 & \text{if } p^f(i) \geq f^\star, \\ 0 & \text{otherwise.} \end{cases}$$

Due to the limited communication range of our agents, $adv(i) = 1$ corresponds to agent i locating prey, large enough for group feeding, and broadcasting this information to its one-hop neighbors. However, since we already assumed that our graph is connected during foraging, the leader agent, l_1, will receive this information as seen in Figure 17. In our framework, once agent l_1 receives the location of prey, it will initiate the capturing phase. In order to capture the fish, it must decide which capturing method to use and which dolphins to employ for that method. The next section will discuss how agent l_1 makes these selections.

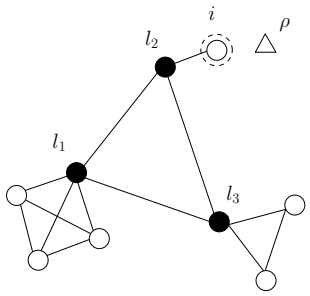


Figure 17: The agents (circles) are searching in groups. The lines denote edges and leaders (shown in black) remain connected during foraging. The agent (i) advertising the centroid of prey (triangle) has a ring around it.

4.2.4 Capture Method Selection Through Voting

As mentioned before, there are five possible methods to capture prey - four variations of the wall method and the carousel method. Let us label these methods for simplicity of notation: "fish in front" as Method 1, "two frontal attacks" as Method 2, "group as a wall" as Method 3, "two columns" as Method 4, and "carousel" as Method 5. Observations presented in [75] show that the number of dolphins participating in these different methods can often be divided into distinct ranges; for example, $5 - 15$

dolphins are usually found in the carousel method. Since agents in our model make decisions based on local interactions, an agent will choose a method based solely on the number of neighbors it has when it receives the location of the prey.

We define the voting profile of agent i as $m(i) \in \mathbb{R}^5$; a column vector with 1 in the row corresponding to the voted method and zero everywhere else. Agent i selects method k, i.e. $m_k(i) = 1$, if $|\mathcal{N}_i| \in [N_k, N_{k+1})$, where $N_k \in \mathbb{N}$, $N_{k+1} > N_k \; \forall \; k \in \{1 \dots 5\}$, and $|\cdot|$ denotes cardinality. Furthermore, we let $N_1 = 1$ and $N_6 = |\mathcal{H}|$ to eliminate the possibility of a no vote.

We still need to establish coordination between agents to avoid a scenario where one group of dolphins is performing a wall method while another group is performing the carousel method to catch the same school of fish. We need a global behavior to emerge from these local opinions and the way we are going to select a suitable method for the entire herd is through a weighted poll conducted by the leader agent, where the weights will correspond to the dominance of each agent.

Let $\bar{m}_k$ denote the votes for method k tallied by the leader agent, l_1. This is calculated as follows:

$$\bar{m}_k = \sum_{i=1}^{N} (d_i \cdot m_k(i)), \; \forall \; k \in \{1 \dots 5\}$$

The leader will direct the herd to use method k if $\bar{m}_k > \bar{m}_n \; \forall \; n \neq k \in \{1 \dots 5\}$. In the event of a tie, the leader selects the method it voted for during the selection process. Once the leader settles on a capturing method, it needs to determine which agents it will allow to participate in the process. The next section discusses how suitable agents are chosen for a particular method through an auction.

4.2.5 Agent Selection Through Auction

As mentioned in [75], not all dolphins are involved in the capturing phase and it is possible that the only the hungriest dolphins are the ones that participate. To model this, we develop an agent selection mechanism once the appropriate method is chosen.

Assume that after the voting process, method k is chosen to capture the prey. As mentioned in the previous sub-section, method k requires $[N_k, N_{k+1})$ agents and to select the appropriate number of agents, we design a simple auction (for more details on auctions, see [69]), where the leader agent is the auctioneer and the prize of the auction is the opportunity to participate in the prey-capturing process.

The bid of each agent for the prize is determined by its "hunger." If b_i represents the bid placed by agent i, then we have $b_i = h_i(t)$, where $h_i(t) \in \mathbb{R}$ is the hunger coefficient of agent i. The hunger coefficient of the agent evolves as

$$\dot{h}_i(t) = \begin{cases} h_{max} - h_i(t) & \text{if } i \text{ not eating fish at time } t, \\ 0 & \text{otherwise,} \end{cases}$$

where $h_{max} > 0$. A first-order ordinary differential equation is chosen for the sake of analytical simplicity. Every time instance that agent i does not capture fish, its hunger coefficient increases; it resets to 0 when it captures fish.

We let v_i be the valuation of the prize by agent i, which represents how much the prize is worth to agent i. We model the valuation as a function that increases with the energy an agent has to exhaust to capture the prey. In our model, the energy exhausted is considered to be a function of the distance the agent needs to travel to start the capturing process. We use a simple formulation for valuation and let $v_i = \alpha \parallel x_i - \rho \parallel$, where x_i is the position of agent i, ρ is the centroid of the school of fish, and α is a scalar.

The utility (or payoff) for agent i is given by $u_i = v_i - b_i$. Since the idea is restrict all the agents from joining the hunting group, we want the agents to first determine whether the hunt is even worth their participation. If the utility is positive, the agents will let the leader know that they are available for the hunt and our agents accomplish this through the function $avail : \mathcal{H} \rightarrow \{0, 1\}$, defined as follows:

$$avail(i) = \begin{cases} 1 & \text{if} \quad u_i > 0, \\ 0 & \text{if} \quad u_i \leq 0. \end{cases}$$

The set of all available agents is denoted by $\mathcal{A}$. Thus, $\mathcal{A} = \{i \mid avail(i) = 1 \; \forall \; i \in \mathcal{H}\}$ represent the agent that are eager to participate in the capture phase of foraging since they will receive a positive payoff. There are three cases that arise based on the number of agents in the set $\mathcal{A}$ and the leader selects agents as follows:

Case 1. $|\mathcal{A}| > N_{k+1}$. In this case, the leader selects the agents from the set $\mathcal{A}$ with the N_{k+1} highest bids.

Case 2. $|\mathcal{A}| \in [N_k, N_{k+1})$. The leader will select all the agents from the set $\mathcal{A}$.

Case 3. $|\mathcal{A}| < N_k$. In this case, the leader will select all the agents from the set $\mathcal{A}$ and to fill N_k positions it will force $N_k - |\mathcal{A}|$ agents with the next highest bids (even if their utilities are negative) to join in the capturing process.

The dominance of agents will be used as a tie-breaker, i.e., for the same bid, the agent with a higher dominance will be selected over the agent with the lower dominance. Notice that in the third case, agents in the set $\mathcal{A}$ benefit from foraging with the herd, which is often the case with social foragers [31].

4.2.6 Foraging Model as a Hybrid Automaton

Since agents might enter or leave the different phases of the foraging process, the system dynamics will undergo discrete transitions. Hence, we model the foraging process as a hybrid automation, as seen in Figure 18. The transition from the state `fuse` to any of the states in the search phase depends primarily on the distribution of predators. The transition to detect phase depends on the distribution of prey and the transition to the capture phase depends on the number of agents available. Also, at anytime during foraging, if the threat level assessed by any group leader is greater than r_{max}, then the agents will abandon the foraging process and return to the state `fuse`.

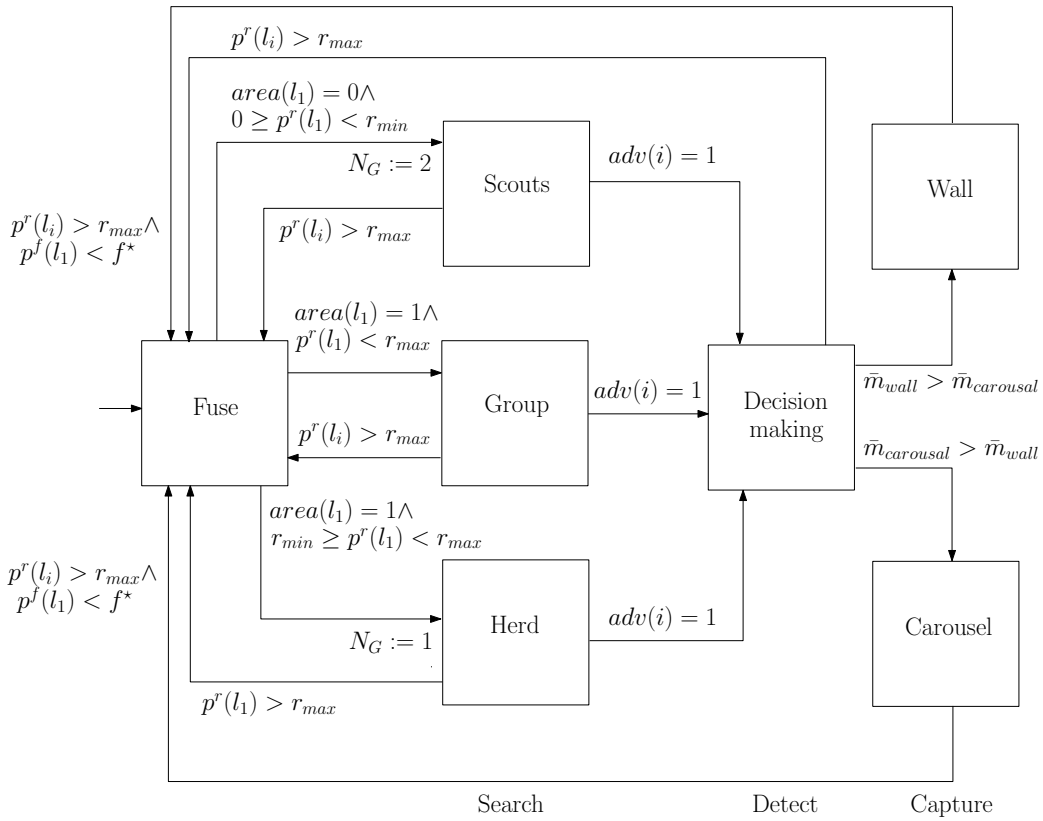


Figure 18: Hybrid automaton model of the entire foraging process.

4.3 Predator and Prey Dynamics

Both predator and prey agents are assumed to have unicycle dynamics:

$$\begin{cases} \dot{x}_i = S_i \cos(\theta_i) \\ \dot{y}_i = S_i \sin(\theta_i) \\ \dot{\theta}_i = \omega_i \end{cases} \tag{14}$$

where $i \in \{f_1, \cdots, f_n, p_1, \cdots, p_N\}$. Indices f_i and p_i represents fish i and predators i, respectively. A fish changes its heading only when it is closer than Δ to other fishes, or when it is inside the *region of influence* of one or more dolphins.

Definition 4.3.1 *The region of influence (ROI) of a dolphin is an arc of radius r and half angle α, with its tip at the current position of the predator and its center line along the velocity vector of the predator.*

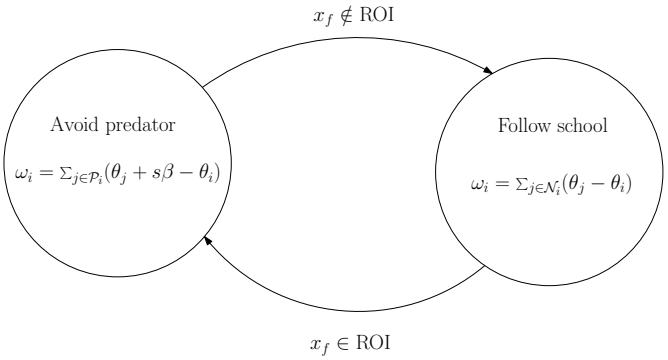


Figure 19: Hybrid automaton governing fish dynamics.

When a fish is inside the ROI of a predator, it tries to escape from the predator by turning towards the perpendicular vector to the velocity of the dolphin. We model this behavior as the fish running consensus protocol on the heading with the heading of the dolphins influencing it plus an angle $\beta \in (0, \frac{\pi}{2}]$. Our simulations show that fixing $\beta = \pi/4$ generates expressive results. When the fish is not inside the ROI of any of the predators, it runs consensus protocol with neighboring fish within Δ distance from it. The latter term models the schooling of the fish. This behavior of the prey constitutes a hybrid automaton depicted in Figure 19 and is driven by

$$\omega_i = \begin{cases} \sum_{j\in\mathcal{P}_i}(\theta_j + s\beta - \theta_i) & \text{if } \mathcal{P}_i \neq \emptyset, \\ \sum_{j\in\mathcal{F}_i}(\theta_j - \theta_i) & \text{otherwise, } \forall\ i \in \{f_1, \cdots, f_n\}. \end{cases} \tag{15}$$

Parameter $s \in \{-1, +1\}$ indicates in which half of the ROI the fish is located. Sets $\mathcal{P}_i$ and $\mathcal{F}_i$ are, respectively, the set of dolphins that fish i is in their ROI and set of other fish that their distance to fish i is closer than Δ.

Motivated by the horizontal carousel formation of the Bottlenose dolphins, we uniformly place them on a circle encompassing the school of fish. They swim counter clockwise with a constant speed, while spiraling toward the center of the circle by reducing their turn radius linearly, i.e.,

$$\begin{aligned} \dot{R}_i &= \begin{cases} -\alpha & \text{if } R_i > R_{min}, \\ 0 & \text{otherwise,} \end{cases} \\ w_i &= S_i/R_i, \qquad \forall\, i \in \{p_1, \cdots, p_N\}. \end{aligned} \tag{16}$$

In three variation of wall method (fish in front, dolphin group as wall, two frontal attack), dolphins form a column and herd the fish toward the shore or another column of dolphins. Heading of the dolphins in these modes is constant, i.e. $w_i = 0$ and they slow down as they get close to the shore or the other column of dolphins.

In the two column variation of wall method a column of dolphin attacking the fish surrounds the school from both sides and then move towards the shore. In our model, a column of dolphins moves towards the school of fish, when they get to a critical distance of the fish centroid, they alternate to go to the right and left side of the school of fish (odds to the right, evens to the left). Then, they adjust their headings to go straight again. They time this action so that their ROI overlaps and covers all the region in between the two column of dolphins. Heading dynamics of the prey in this mode is expressed as

$$\begin{aligned} w_{p_1}(t) &= \begin{cases} S_p/R & \text{if } t < t_{c_1}, \\ -S_p/R & \text{if } t_{c_1} \le t < t_{c_2}, \\ 0 & \text{otherwise,} \end{cases} \\ w_{p_i}(t) &= w_{p_1}(t + (i-1)\eta). \end{aligned} \tag{17}$$

where R is their turn radius and η is the delay in the execution of the command. Parameters t_{c_1} and t_{c_2} characterize the time it takes to surround the school of fish and the time it takes to go back to the straight swim and are determind by the speed and turn radius of the dolphins to ensure there is no gap in the coverage of ROIs between the two formed columns of the dolphin.

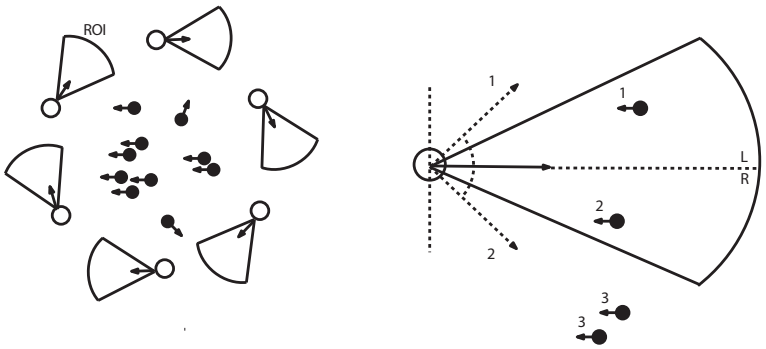


Figure 20: *Left:* The region of influence (ROI) of the dolphins (white) are shown. *Right:* Fish (shown in black) inside the ROI run consensus with the projected dolphin heading, depending on which side of the ROI they lie (fish heading 1 is running consensus with dolphin heading 1). Fish outside the ROI run consensus only with their neighbors in the school.

4.4 Simulations

We randomly initialize a group of 30 fish inside a disk or radius 30. In the horizontal carousel mode, dolphins start the encirclement on a circle of radius 50 around the fish. We assume that the ROI of a dolphin has radius $r = 10$ and half angle of $\alpha = .3\pi$. Figure 21(left) depicts the evolution of the system. This figure shows how fish get entrapped inside the dolphins carousel. The same model provides a rich enough to mimic variations of the wall method as well. Figure 21(right) illustrates the "two column" variation of the wall method and Figure 22 illustrates the "group as wall" variation.

4.5 Formation Selection During Foraging

The dolphin dynamics in the horizontal carousel method assumed that the dolphins were uniformly placed on a circle centered at the centroid of the school of fish, ρ. As a result, an uncoordinated, spiraling motion was able to maintain the encirclement around the centroid. However, if the dolphins are not uniformly placed, they need to coordinate with each other in order to maintain an encirclement around the fish. This problem is simplified to a scenario where each agent has a choice between executing

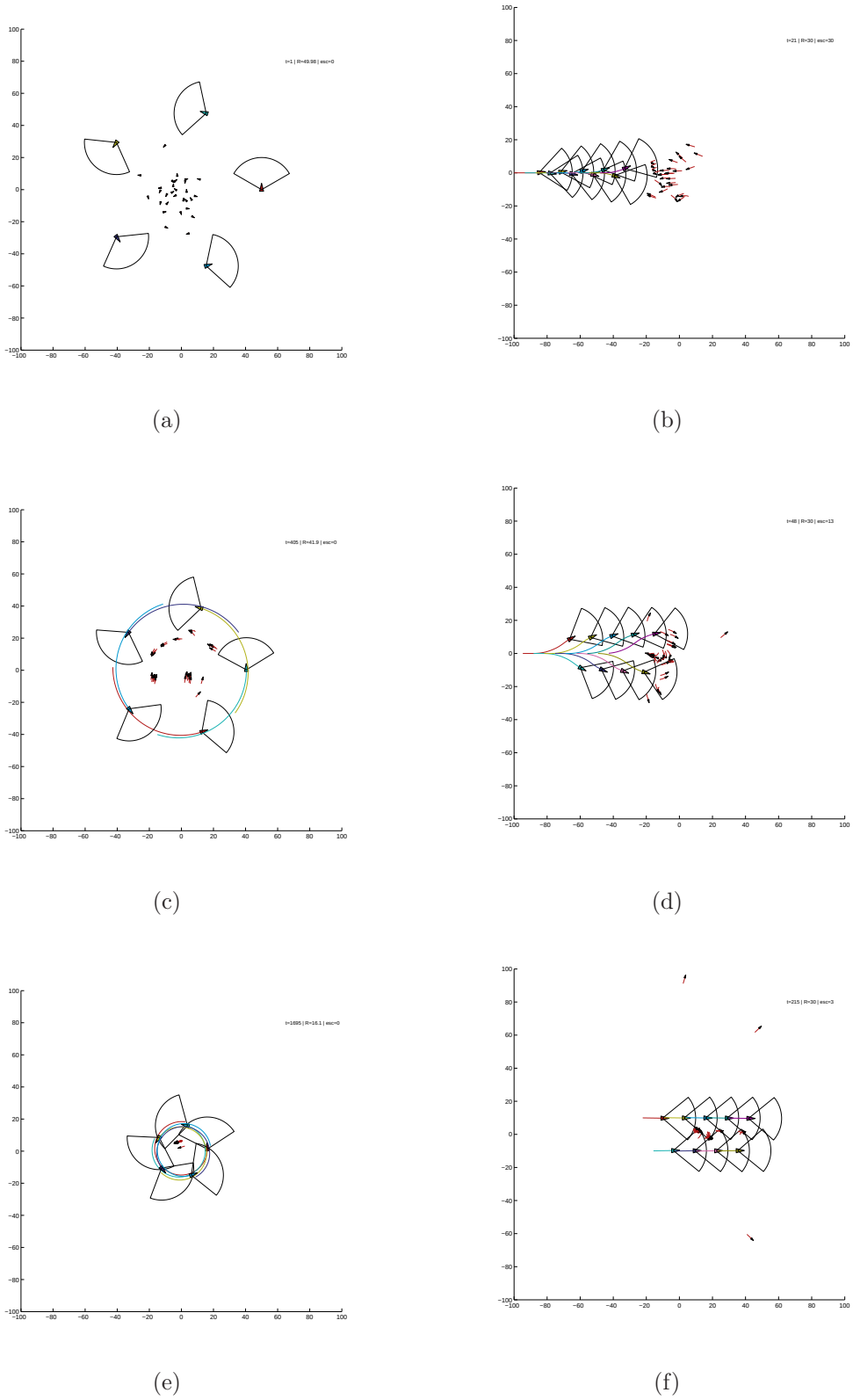
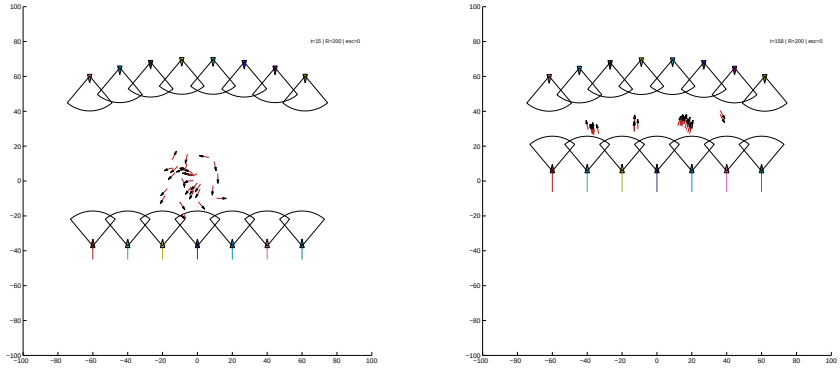


Figure 21: School of fish entrapped in the Bottlenose dolphins horizontal carousel (left) and two column wall method (right).



(a) (b)

Figure 22: Dolphins feed on the fish bounced back from the shore or another column of dolphin acting as wall.

two formations, namely, Large and Small circle.

Although there exists many formation control strategies for autonomous agents ([22] and [94] are two examples), there does not exist any underlying theory regarding the selection of formations. In essence, any formation selection mechanism is a hybrid control strategy. The selection in this approach is driven by the agents' estimates of their errors associated with each formation; consequently, the formation with the least error is selected. Three selection strategies are presented here and the advantages and disadvantages, in terms of complexity, communication, and extraction of global properties, are presented for each strategy (for details, see [33]).

4.5.1 Strategy 1: Local Instantaneous Errors

Let $E_{i,1}(t)$ and $E_{i,2}(t)$ be the local instantaneous error of performing Large circle and Small circle, respectively, for agent i. The local instantaneous error is defined as

$$E_{i,j}(t)^2 \;=\; \sum_{k \;\mid\; (i,k)\in\mathcal{E}} ((\parallel x_i(t) - x_k(t) \parallel - K_j)^2 + (\parallel x_i(t) - \rho \parallel - r_j)^2), \qquad (18)$$

$\forall\, i \in \mathcal{H},\ \forall\, j = \{1, 2\}$. To perform formation j, each agent keeps track of the distance to its one-hop neighbors (K_j) and the distance to the centroid of the fish (r_j). The

selection mechanism proposed under this strategy is given by Figure 23.

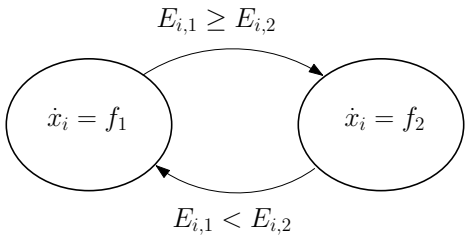


Figure 23: In Strategy 1, agent i selects the dynamics $\dot{x}_i(t) = f_1$ (form Large circle) over $\dot{x}_i(t) = f_2$ (form Small circle) if $E_{i,1} < E_{i,2}$.

The disadvantage of this strategy is that the decisions are made based solely on local properties and an agent does not consider the decisions of other agents in the network. Nonetheless, the advantage of this strategy is that each agent is only required to contain the relative displacements with its one-hop neighbors.

4.5.2 Strategy 2a: Instantaneous Averaged Initial Errors

Let $\xi_{i,j}(t)$ be agent i's estimate of the global error associated with formation j, defined as

$$\dot{\xi}_{i,j}(t) = - \sum_{k \mid (i,k) \in \mathcal{E}} (\xi_{i,j}(t) - \xi_{k,j}(t)), \tag{19}$$

$\forall\ i \in \mathcal{H},\ \forall\ j = \{1, 2\}$, and $\xi_{i,j}(0) = E_{i,j}(0)$. The global error estimate of an agent is the average of the initial local errors of itself and its neighbors. The proposed selection strategy is that agent i executes $\dot{x}_i(t) = f_1$ if $\xi_{i,1}(t) < \xi_{i,2}(t)$ and $\dot{x}_i(t) = f_2$ if $\xi_{i,1}(t) \geq \xi_{i,2}(t)$.

In this strategy, the decisions of other agents in the network are taken into account; as a result, agents must be capable of communicating with other agents instead of only measuring relative displacements, as seen in Strategy 1.

4.5.3 Strategy 2b: Delayed Averaged Initial Errors

Strategy $2b$ is Strategy $2a$ without instantaneous switches. Here, agents select a particular formation once (19) settles below some threshold. A method to calculate

the convergence time, T_{conv}, is presented in [68].

In this strategy, agents are required to communicate with other agents and measure the convergence time. Also, the global performance estimates of an agent is based on the initial configuration of the network and any new information that might be available in the network does not factor into the selection strategy.

4.5.4 Strategy 3: Dynamic Averaged Errors

In this strategy, the global performance estimates are driven by the injection of new information by using dynamic average consensus estimators developed in [28]. The global estimate is given by

$$\dot{\xi}_{i,j}(t) = - \sum_{k \mid (i,k)\in\mathcal{E}} ((\xi_{i,j}(t) - \xi_{k,j}(t)) + F(E_{i,j}(t) - \xi_{i,j}(t))), \tag{20}$$

$\forall\, i \in \mathcal{H}$, $\forall\, j = \{1, 2\}$, and $F(E_{i,j}(t) - \xi_{i,j}(t))$ is an insertion of the instantaneous local error, $E_{i,j}(t)$ (the new information available in the network). There are two ways to insert new information into a dynamic consensus estimator and the details can be found in [33].

This strategy is computationally more complex than all the previous strategies. However, it incorporates any new information that might be available in the network.

4.6 Concluding Remarks

We produced a hybrid automaton of the foraging behavior of Bottlenose dolphins in the context of heterogenous multi-agent herding. Simulation results showed that the simple model developed here is expressive enough to capture the foraging techniques used by dolphins. Also, based on the horizontal carousel method to catch fish, we presented three decentralized formation selection strategies and discussed the advantages and disadvantages associated with each strategy.

CHAPTER V

GROUP SIZES

In this final chapter on producing agent-based models of biological systems, we focus on African lion prides to address the issue of maintaining sustainable group sizes in multi-agent systems. The main contribution of this chapter is the following: for a given number of males and females, we determine whether the group is sustainable and by sustainable, we mean that females can forage sufficient prey to feed the entire pride and at the same time, there are an adequate number of males to patrol the territory.

5.1 Introduction

Although problems such as formation control, consensus, and containment have been exhaustively studied in the field of multi-agent systems (for a representative sample, see [45, 60, 67, 78]), they typically involve issues pertaining to multi-agent classifications such as heterogeneity, distributiveness, and communication. The size of a multi-agent system is rarely addressed outside of the multi-robot foraging literature.

Foraging, as understood in the multi-robot context, involves agents that search and contain objects in the environment, with applications found for example in search and rescue scenarios. For pure foraging tasks, the issue of sustainable sizes for a multi-agent system has been previously addressed (for a representative sample, see [3, 56, 70]) and the idea is to let the number of agents be selected based on social foraging theory, in which the performance of a group increases with size until a critical number is reached [70].

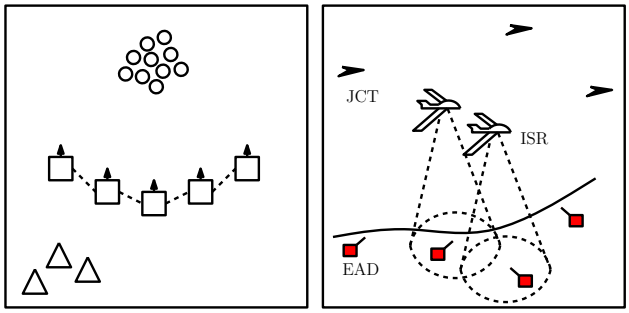
This idea is applicable only if *all* the agents are performing the same task. The

novelty of this work is that we determine sustainable sizes for a heterogeneous multi-agent system that consists of a foraging team together with a team providing boundary protection by patrolling the perimeter of the foraging area. To find sustainable sizes for such engineered systems, we draw inspiration from natural systems; in particular, we consider the social structures of African lions.

A pride of African lions contains up to 9 males and 18 females [23]. And as mentioned in Chapter 2, if the size of the pride grows too large, members often break away and factors such as food availability and the ability to ward off intruders influence the size of a pride [84]. Females are usually in charge of foraging for food, while males are responsible for territorial defense. With this biological system as our inspiration, we analyze sustainable sizes for a multi-agent system by producing a simple, yet expressive model that can be applied to engineering applications such as search-and-patrol using teams of autonomous vehicles.

A potential application of our work is determining the sustainable sizes for the US Navy suppression of enemy air defenses (SEAD) mission conducted by teams of autonomous vehicles. This mission will be discussed in more details in Chapter 8, but briefly, the mission involves gathering intelligence on the enemy's air defenses (e.g., nature, location, etc.) by a team of UAVs known as the Intelligence, Reconnaissance, and Surveillance (ISR) team, while another team known as the Joint Combat Team (JCT) maintains air supremacy for the ISR team [1] (Figure 24(b)). Recall that the goal is not biomimicry, i.e. to replicate all aspects of the natural system; instead, the goal is to extract characteristics and draw on biology for engineered applications. However, it is important for our model to be rich; in fact, we will show that when we apply biological field data, such as the encounter rate with prey, our sustainable group sizes closely resemble actual pride sizes.

The remainder of the chapter is organized as follows: In Section 5.2 we present our model of the pride that is used to determine a set of sustainable pride sizes and



(a) (b)

Figure 24: In (a), the pride structure is shown; males (triangles) patrol the pride area, whereas female lions (squares) are in charge of foraging for prey (circles). In (b), a search-and-patrol application is shown where combat teams (labeled JCT) provide security for surveillance teams (labeled ISR) that are searching for vehicles (labeled EAD).

in Section 5.3, we introduce a notion of utility for a group and identify the optimal group size. Finally, conclusions are presented in Section 5.4.

5.2 Pride Model

According to our notion of sustainability, a sustainable group size should successfully forage enough food for the entire group and at the same time provide territorial defense. We will present metrics for both these tasks and provide simulations for different values of the parameters used in this model, such as the number of females a male will consort at a time.

5.2.1 Foraging

Assume that a pride consisting of M males and F females requires a minimum of $P_{min}(M, F)$ of food to survive. If a lion and a lioness needs an average of p^M and p^F kg/hr of food, respectively, then the minimum energy intake required by the pride is $P_{min}(M, F) = Mp^M + Fp^F$ kg/hr (typically, $p^M = 2.5$ and $p^F = 2.2$ according to [84]) .

We will assume that only lionesses are responsible for foraging and hunting prey.

Lionesses are capable of hunting small prey like warthogs (*Phacochoerus aethiopicus*) and wildebeest (*Connochaetes taurinus*) on their own. However, their success rate is low with larger prey, such as zebra (*Equus burchelli*) and buffalo (*Syncerus caffer*); in fact, hunting a large prey like a buffalo requires males to join the hunting group as well [73], [29]. We assume that the pride is specifically hunting zebras, a medium sized prey that is large enough to require a group of females, but small enough to exclude the participation of males.

The success rate of capturing prey increases with group size if the success rate of a lone hunter is low [85]; however, a foraging group too large becomes conspicuous to prey and takes away its ability to stalk prey [23]. The type of prey lions hunt can out run them [23] and this is why "stalking" prey is important to a successful hunt.

Let $Pr(F) \in [0,\ 1]$ be the probability of an encountered prey being captured when there are F females in the pride. $Pr(F)$, inspired by the group performance curve in social foraging theory [31], is quadratic in F with maximum value at $F = F'$. More specifically,

$$Pr(F) = \begin{cases} \frac{F(2F'-F)-(2F'-1)}{(F')^2-2F'+1} & \text{if } 1 < F < 2F' - 1, \\ 0 & \text{otherwise,} \end{cases} \tag{21}$$

where F' is the optimal number of hunters. This is a simple formulation that captures the following key ideas: a single lioness cannot capture a medium sized prey, the amount of prey caught generally increases with the number of hunter, and yet too many hunters takes away their ability to remain inconspicuous to their prey (typically $3 - 8$ lionesses are observed in the hunt according to [23]).

Table 5 in [84] lists the encounter rate with zebras for Serengeti lions during both prey scarcity and abundance. The encounter rate, λ, is measured in a 2000 km^2 area and has the unit $herd/hr$. For that area, $\lambda = 0.008$ $herd/hr$ represents a scarce zebra density and $\lambda = 0.245$ $herd/hr$ was considered an abundant zebra density. The number of individuals in a herd and the average weight of an individual prey is

also provided in [84] and with these data, we can calculate λ in units of $kg/km^2/hr$. With this formulation, the encounter rate with zebra would become $\lambda \in [0.8, 24.6]$ $kg/km^2/hr$.

We assume that the foraging area is a circle with radius $R(M, F)$ and thus, for a given number of females, F, and encounter rate, λ, the expected amount of prey captured is

$$P_{cap}(F) = Pr(F)\lambda\pi R(M, F)^2, \tag{22}$$

measured in kg/hr. If we assume that the lionesses select the size of the foraging area such that $P_{cap} = P_{min}$, then the radius of the foraging area is given by

$$R(M, F) = \sqrt{\frac{P_{min}(M, F)}{Pr(F)\lambda\pi}}, \tag{23}$$

measured in km and a size is sustainable from a food availability point of view if $0 < R(M, F) < \infty$. Even if a pride is sustainable energy-wise, it might still be defense-wise unsustainable. Next, we look at the factors that determine the ability for males to protect the pride from potential intruders.

5.2.2 Territory Defense

We described how the lionesses specify the radius of the foraging area, $R(M, F)$, to meet the energy demand of $P_{min}(M, F)$. If we assume that the foraging area set by the females is indeed the pride area, then males need to defend a circle with radius $R(M, F)$ from intruders. Furthermore, we will also assume that the males reside in the boundary of this circle, equidistant from each other.

Lions communicate through roars, that can be heard about 8 km away during territorial advertisement and when intimidating intruders [23]. With this notion of a limited communication range of a lion, we can define the minimum number of males needed to patrol the pride area, M^d_{min}, as follows:

$$M^d_{min}(R) = \frac{2\pi R(M, F)}{\Delta},$$

where Δ is the maximum allowable arc length between two males that prevents intruders from entering into the pride area and the subscript "d" is used to denote defense. Note that our notion of territory defense does not depend on actual confrontation with intruders, rather it depends on the ability of the males to "plug holes" in the boundary of the pride area.

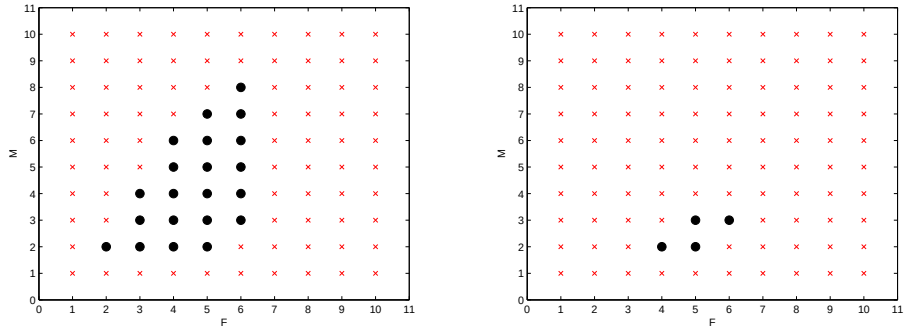
In a pride, males are also competing with each other to mate with females and we model this "constant competition" [23], or in-fighting, by assuming that each male consorts k females and each non-consorting male is involved in a fight with a consorting male. According to [72], there can be "serious fights" between consorting and non-consorting males and in our model, we regard such in-fighting among resident males as a distraction from their primary role of patrolling the pride area.

We let the number of males patrolling the pride be given by the function $G(M, F)$, and only require that $G(M, F) = M$ when $M \leq F/k$ and that $G(M, F)$ is decreasing with increasing $M - F/k$. One formulation that satisfies both of these requirements is

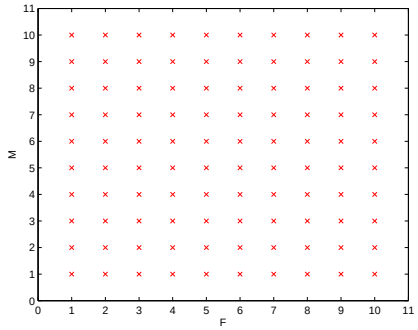
$$G(M,F) = \begin{cases} M & \text{if } M \leq \frac{F}{k}\ , \\ M - 2(M - \frac{F}{k}) & \text{otherwise.} \end{cases}$$

A pride size is sustainable from a territory defense perspective if $G(M, F) \geq M^d_{min}(R)$. Finally, a group size, which we denote by the ordered pair (M, F), is considered sustainable if it is both energy-wise and defense-wise sustainable. More precisely, the set $\mathcal{S} = \{(M, F) \mid 0 < R(M, F) < \infty,\ G(M, F) \geq M^d_{min}(R)\}$ denotes the set of sustainable group sizes.

Within the sustainable set of prides, a smaller radius is more desirable by the lionesses since a smaller area reduces encounters with intruders, which will in turn ensure more safety to their cubs [23, 41]. Also, within the set of sustainable prides, more patrolling males will guarantee more "cushion" from non-resident males; thus, it is likely that the biological system itself has a preference on the size of the group,



(a) $k = 1$ (b) $k = 2$

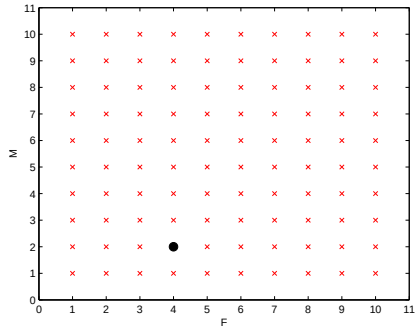


(c) $k = 3$

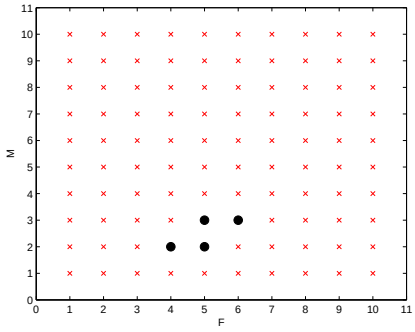
Figure 25: Sustainable group sizes for different values of the number of females a male can consort at a time, k, are denoted with an dot, while unsustainable group sizes are denoted by a cross. In (a)-(c), the values of other parameters are as follows: $\lambda = 6$, $F' = 4$, and $\Delta = 10$.

but since we intend the artificial systems to draw from nature (and not the other way around), in the next section, we address this idea of assigning values to group sizes within the sustainable set in the context of engineered systems.

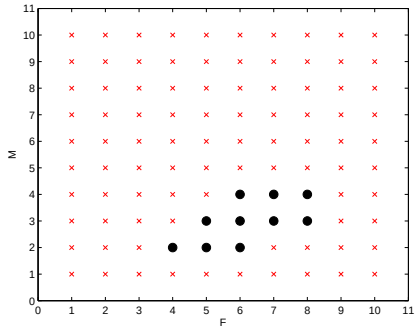
Simulations based on this model are shown in Figures 25-27. In Figure 25, the effects of varying the number of females each male consorts, k, on the sustainable sizes are shown. As k increases, each male guards more females and this increases the number of non-consorting males in the pride. Thus, in-fighting increases with k, which distracts more males from patrolling the territory and makes the pride more susceptible to intruders. As a result, the set of sustainable prides decreases as k increases, when λ, F', Δ are all held constant.



(a) $F' = 3$

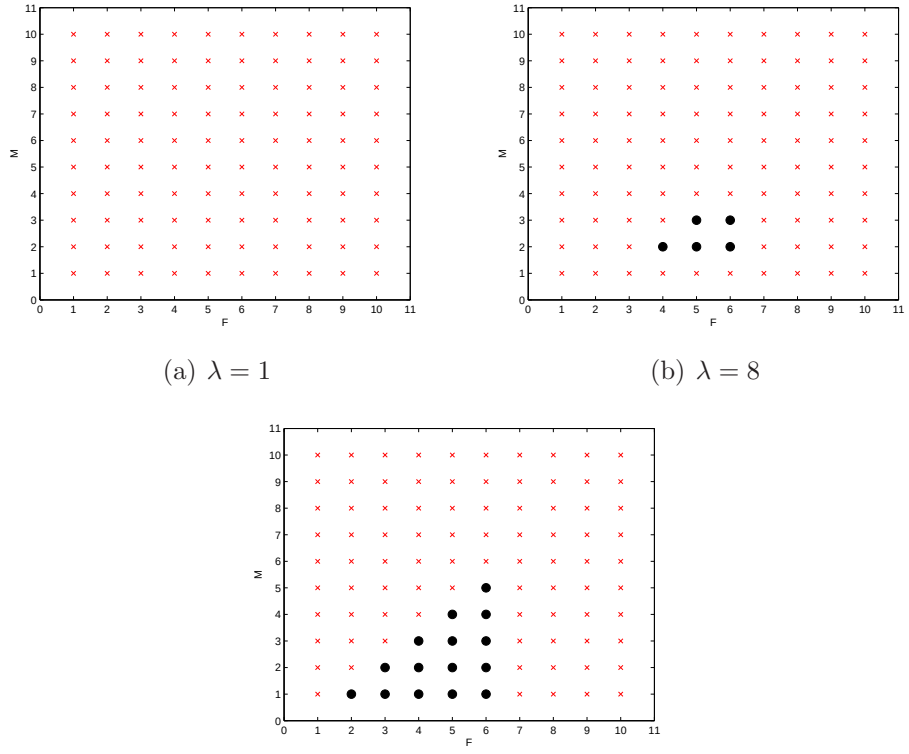


(b) $F' = 4$



(c) $F' = 5$

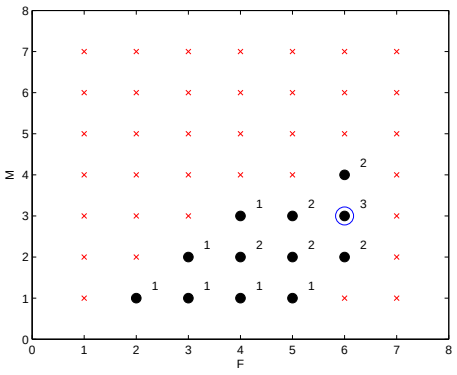
Figure 26: Sustainable group sizes for different values of the optimal number of foragers, F', are denoted with an dot, while unsustainable group sizes are denoted by a cross. In (a)-(c), the values of other parameters are as follows: $\lambda = 6$, $k = 2$, and $\Delta = 10$.



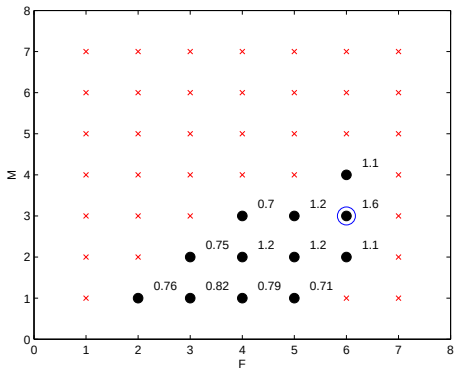
(a) $\lambda = 1$ (b) $\lambda = 8$

(c) $\lambda = 24$

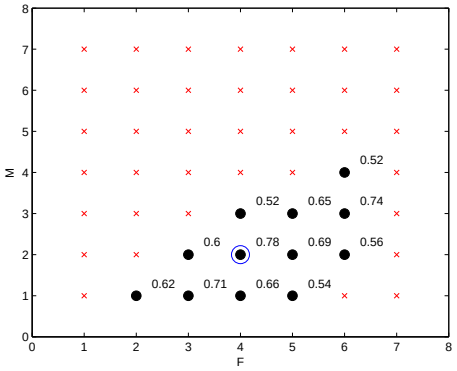
Figure 27: Sustainable group sizes for different values of the encounter rate with prey, λ, are denoted with an dot, while unsustainable group sizes are denoted by a cross. In (a)-(c), the values of other parameters are as follows: $F' = 4$, $k = 2$, and $\Delta = 10$.



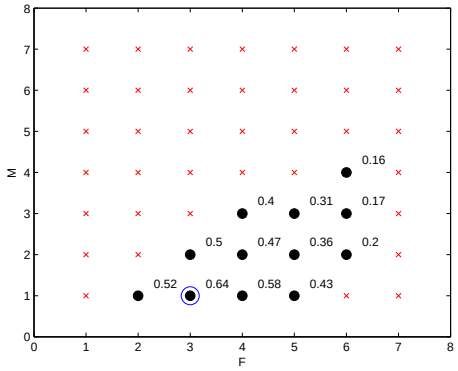
(a) $\omega = 0$, $(M^\star, F^\star) = (3, 6)$



(b) $\omega = 0.5$, $(M^\star, F^\star) = (3, 6)$



(c) $\omega = 0.8$, $(M^\star, F^\star) = (2, 4)$



(d) $\omega = 1$, $(M^\star, F^\star) = (1, 3)$

Figure 28: The optimal group size (ring) is shown for different values of ω. The sustainable group sizes are denoted with a dot, while unsustainable group sizes are denoted by a cross. In (a)-(d), $\lambda = 15$, $k = 2$, $F' = 4$, and $\Delta = 10$.

In Figure 26, the effects of varying the optimal number of foragers, F', is shown. As F' increases, from (21) we notice that the support of the function $Pr(F)$ increases; thus, the set of sustainable group sizes increases too. The encounter rate with prey, λ, is varied in Figure 27. For a large λ, the foraging radius is small and as a result, the foragers can meet the energy intake for larger prides compared to those under smaller values of λ. Our group sizes mostly consist of $1-3$ males $2-9$ females, using field data recorded in [23, 84, 85] for λ, F', k, and Δ. Thus, our simple mathematical model is in fact expressive enough to capture the underlying structures of lion prides.

In the next section, we assign a utility to each sustainable group size and consequently, identify an optimal group size.

5.3 Optimal Group Size

In the previous section, for a given group size (M, F) we developed a method to characterize it based on parameters like the communication range of males, Δ. For engineered systems, it may also be useful to know the optimal group size, $(M^\star, F^\star)$, from the set of sustainable sizes. We define a utility function based on our multi-agent search-and-patrol task as follows:

$$U(R, G) = \omega \frac{1}{R(M, F)^2} + (1 - \omega) G(M, F), \tag{24}$$

where $\omega \in [0, 1]$. With this particular choice of a utility function, when $\omega = 1$, the optimal size minimizes the radius of the foraging area for lionesses and for a group of UAVs, this corresponds to the size that minimizes the radar footprint of the joint ISR and JCT fleet described in Figure 24(b). Also, $\omega = 0$, could correspond to the scenario that requires the JCT team to provide the maximum possible security to the ISR team. Optimal group sizes for different values of ω are shown in Figure 28.

Recall that $\mathcal{S}$ is the set of ordered pairs that represent the sustainable group sizes. Given $\mathcal{S}$, we define $\mathcal{M} = \{M \mid (M, F) \in \mathcal{S}\}$, i.e. the projection of $\mathcal{S}$ onto the first coordinates (males) and define $\mathcal{F} = \{F \mid (M, F) \in \mathcal{S}\}$, i.e. the projection of $\mathcal{S}$ onto

the second coordinates (females). With this notation, we are now ready to present the results of this paper:

Lemma 5.3.1 *(Maximum security): If $\omega = 0$ in the utility function given by (24) and we denote F_{max} as the largest element in $\mathcal{F}$, then the optimal group size is $(\frac{F_{max}}{k},\ F_{max})$.*

Proof If $\omega = 0$ in (24), then the utility function is given by $U(R, G) = G(M, F)$. For a given size $(M, F) \in \mathcal{S}$, the function $G(M, F)$ is is maximized when $M = \frac{F}{k}$, as shown in Figure 29. As a result, $U(R, G)$ is maximized over $\mathcal{S}$ when $F = F_{max}$ and $M = \frac{F_{max}}{k}$, where F_{max} is the maximum element in $\mathcal{F}$.

Lemma 5.3.2 *(Minimum footprint): If $\omega = 1$ in the utility function given by (24), and we denote M_{min} as the smallest element in $\mathcal{M}$, and let $\tilde{F}$ minimize $\frac{P_{min}(M_{min},F)}{P_{cap}(F)}$, then the optimal group size is $(M_{min}, \tilde{F})$.*

Proof If $\omega = 1$ in (24), then the utility function is given by $U(R, G) = \frac{1}{R(M,F)^2}$. For two sustainable sizes (M_1, F) and (M_2, F), it is obvious that $R(M_1, F) < R(M_2, F)$ if $M_1 < M_2$. Thus, for the optimal group size $(M^\star, F^\star)$, $M^\star = M_{min}$, where M_{min} is the smallest element in $\mathcal{M}$ and from the definition of $P_{cap}(F)$ in (22), $U(R, G)$ is minimized over $\mathcal{S}$ when $F^\star$ minimizes $\frac{P_{min}(M_{min},F)}{P_{cap}(F)}\ \forall\ F \in \mathcal{F}$.

5.4 Conclusions

We developed a simple mathematical model of the African lion pride. When biological field data were used, our model produced sustainable sizes that were consistent with actual pride sizes. A possible application of the work is the characterization of sustainable sizes for a multi-agent system that consists of two classes of agents: one class is responsible for searching an area; the other for providing perimeter security for that area.

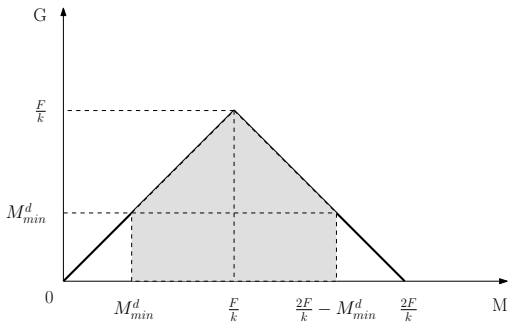


Figure 29: A plot of the function, $G(M, F)$, which represents the number of patrolling males is shown.

CHAPTER VI

CONFINEMENT STRATEGIES

With this chapter, we begin the second phase of our work: the development of bio-inspired algorithms. As mentioned in Chapter 1, the problem class directly influences the outcome of this particular phase of our work. Here, we address the confinement of a group of mobile robots and the confinement algorithms developed in this chapter are based on the foraging model of Bottlenose dolphins generated in Chapter 4. For a multi-agent system, we achieve the following goals: i) provide an algorithm for one group of agents to perpetually confine the other group and ii) characterize the regions from which the herded agents are guaranteed to be captured.

6.1 Introduction

Confinement of mobile robots using decentralized algorithms is of significant interest to the robotics community and an enabling technology for a number of proposed engineered applications [2]. There are usually two classes of agents and the goal of one class, namely the foragers, is to confine the agents from the other class, the herded agents. Previous research in this area focused on cases where the foragers either confine passive objects, i.e. gather them in a specified location [5, 52, 70, 96], or they confine active agents that cooperate with the herders, as shown in [25]. But what if these active agents are non-cooperative?

Using the agent-based model of Chapter 4, this work explores the effectiveness of prey capturing techniques employed by Bottlenose dolphins as the capability of the prey increases. In Chapter 2, we discussed the intelligence of porpoises - the family of dolphins and whales - and presented the coordinated methods known as the *wall method* and *horizontal carousel method* with which dolphins capture fish. In

this chapter, first, inspired by horizontal carousel method of circling fish, we present conditions under which dolphins (or porpoises) can perpetually confine a school of fish in a specified radius of confinement. For this confinement algorithm, we assume that when a fish agent senses a dolphin agent (enters a region near the dolphin), it will turn to escape the predator. Furthermore, we assume that the dolphin agents are aware of this evasive behavior, which is in fact similar to the collision avoidance maneuvers employed by unmanned vehicles. Based on this simplistic behavior of prey, we present conditions under which the predator agent can "bounce" prey agents in a pre-specified region, and in the process, play what we refer to as perpetual porpoise ping–pong (P^4). Thus, we begin with a simple model of prey behavior, and then investigate the effects of optimally moving prey. Consequently, we identify the regions, zones of no escape (ZONE), from which fish cannot escape despite their best effort.

This chapter is organized as follows: a modified dolphin-fish interaction model, from that of Chapter 4, is presented in Section 6.2. In Section 6.3, we provide analytical results of playing different porpoise ping–pong games. The regions of guaranteed no escape for fish are identified in Section 6.4 and concluding remarks are presented in Section 6.5.

6.2 Predator-Prey Interaction

The dolphin foraging process was modeled as a hybrid automaton in Chapter 4 to prescribe rules for searching fish and selecting between the wall and the carousel method. There, a dolphin-fish interaction rule was presented that captured the fleeing behavior of fish in the presence of predators by defining a region of influence (ROI). When a fish is inside the ROI of a predator, it attempts to escape by turning away from the heading of that predator (Figure 20). During P^4, the goal is to confine fish and as a result, we will use the ROI to model the predator-prey interaction, but assume that fish bounce off of the boundaries of an ROI, i.e., they instantaneously

attempt to escape as soon as they sense the predator. This simulates a "bounce" off the ROI and the reflection rules for such a bounce is discussed in the next section in more detail. However, for specifying the ZONE, where the goal is to identify the regions from which fish are guaranteed to be caught, we will require a notion of fish being captured. Consequently, we define a *kill zone* (Figure 30).

Definition 6.2.1 *Kill zone is a triangle with sides r_k and base height b_k. A fish is not successful in its escape if it resides in a kill zone associated with the dolphins.*

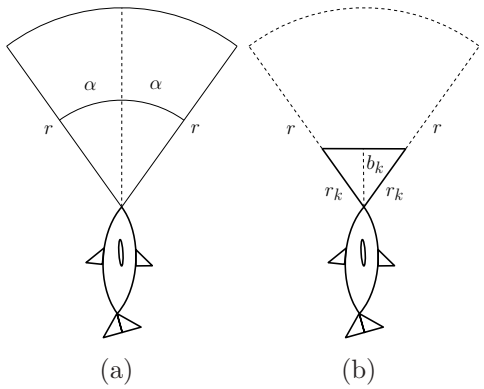


Figure 30: (a) Region of influence (ROI) of Chapter 4 is defined by the radius r and half angle α. (b) The kill zone (solid) is shown with respect to the ROI.

It is obvious that the efficiency of a confinement strategy depends on the prey dynamics (a slower moving prey is clearly easier to catch than a more agile one). To explore dolphin-inspired confinement strategies in the multi-robot context, we choose to study the effectiveness in different settings as the capability of the prey increases. As such, we begin with a simple model of prey behavior, where we assume that fish agents are reactive and incapable of devising sophisticated evasive actions; in fact, they simply "bounce" off the boundary of ROI. Furthermore, we assume that the prey agents move with constant velocities and follow strict reflection rules when sensing predator agents. Such a navigation strategy, although simple, is utilized in many service applications, such as robotic vacuum cleaners [36]. Markers equipped with

infrared signals are used to confine these devices to specific areas and prevent them from driving through doorways. Thus, using the P^4 algorithm (presented in the next section), we can replace the markers needed in an area with a single mobile marker (dolphin agent) capable of pulsing an infrared signal (ROI). We consider this prey behavior to be at one end of the prey capability spectrum - the least capable prey. We will, in subsequent sections, move to the other end of the spectrum and investigate the effects of optimally moving prey.

6.3 Perpetual Porpoise Ping-Pong (P^4)

In this section, we identify the necessary condition for perpetual confinement of mobile robots by a more agile robot inspired by the horizontal carousel maneuver of porpoise. We first consider the confinement of one fish by a dolphin and later extend this to the case of multiple fish.

Assume the first contact of the fish with the boundary of the ROI is at the point P with distance l from the base of ROI. Our goal is to determine the speed of dolphin to make sure the fish will hit the same point on ROI when they meet again. We assume the fish bounces off the boundary of the ROI with a constant angle β regardless of angle of initial impact. The reason for this is that it serves as a starting point for our investigation and we will later enhance the model to follow Snell's law of reflection.

To refer back to the robotic vacuum cleaners, as artificial systems, their response to the infrared signals is part of the design process and the reflection strategies investigated here can thus be easily implemented on such systems. To this end, for simplicity, we first assume that the reflection angle is constant and later, we analyze the case where the prey bounces follow Snell's law of reflection.

Definition 6.3.1 *Perpetual confinement is the confinement of fish by a single dolphin when the fish stays within the region of confinement by hitting the same point on the boundary of ROI perpetually.*

We denote the radius of the circle whose perimeter is patrolled by the dolphin agent by R and assume constant speed for the fish v_f, while

$$R_b = \sqrt{R^2 + l^2 - 2Rl\sin(\alpha)}$$

represents the radius of the region of confinement and is the distance between the center of the circle traveled by the dolphin to the point of impact on the boundary of ROI, P. As shown in Figure 31, the angle between the position of the dolphin and point P is denoted as

$$\delta = \cos^{-1}\left(\frac{R^2 + R_b^2 - l^2}{2RR_b}\right).$$

Proposition 6.3.1 *In the case with a single dolphin confining a fish, in order to achieve perpetual confinement, the dolphin should set its velocity to*

$$v_d = \left(\frac{R}{R_b}\right)\left(\frac{\vartheta}{\sin(\vartheta)}\right)v_f, \tag{25}$$

where $\vartheta = \beta - \delta$.

Proof From the geometry of the problem the time it takes the fish to travel before bouncing again is $2R_b\sin(\vartheta)/v_f$ while the dolphin travels a distance of $2R\vartheta$ at the same time.

Corollary 6.3.1 *In case of one fish and one dolphin a constant velocity for the dolphin will guarantee perpetual confinement.*

Proof As the fish bounces off the same point on ROI, all the variables in (25) stay constant, rendering the velocity of dolphin to be constant.

Alternatively, we can parameterize the meeting points with each fish by the time to meet and the position of dolphin on the patrol circle (parameterized by dolphin's heading), i.e., (t_i, Θ_i). Then dolphin agent's task is to adjust its speed to meet the

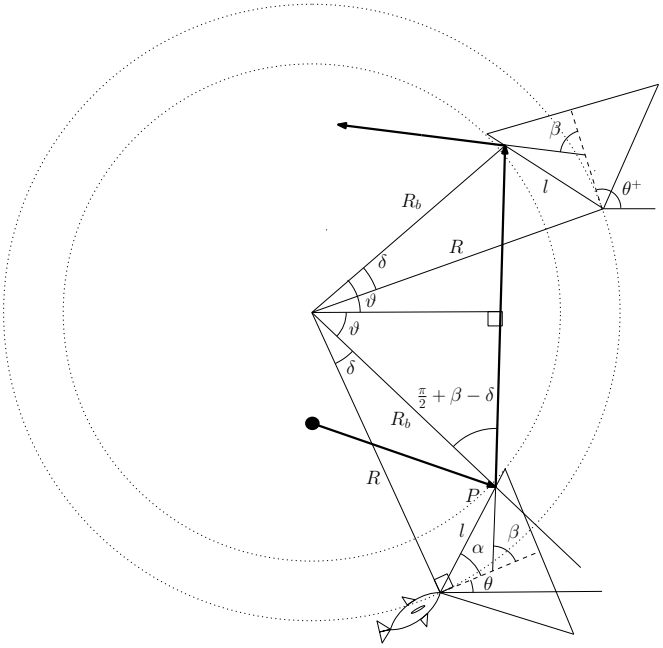


Figure 31: Using the geometry of the problem we can calculate the speed of the dolphin such that the fish (full circle) hits the boundary of ROI at the same point at each bounce.

fish at the specified time and position. This provides a framework for perpetual confinement of multiple fish, where a sorted list of (t_i, Θ_i) determines which fish should be met next.

After the i-th fish is bounced, the tuple (t_i, Θ_i) is computed (and stored) as

$$(t_i, \Theta_i) = (t + 2R_{bi}\sin(\vartheta_i)/v_f, \theta + 2\vartheta_i), \tag{26}$$

where R_{bi} and ϑ_i are R_b and ϑ, respectively, calculated for fish i. The velocity of dolphin is set to

$$v_d = \begin{cases} 2R(\Theta^* - \theta)/(t^* - t), & \text{if } \Theta^* - \theta > 0 \\ 2R(2\pi + \Theta^* - \theta)/(t^* - t), & \text{otherwise;} \end{cases} \tag{27}$$

where Θ^* represents where the dolphin should be at t^* to bounce the next fish. Figure 32 presents the perpetual confinement simulation where a dolphin confines a group of three fish using the scheduling scheme described earlier.

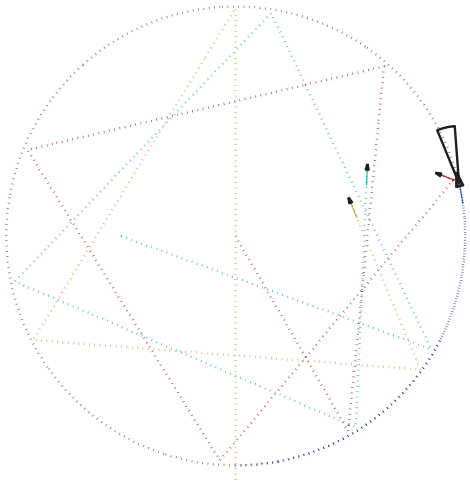


Figure 32: Perpetual confinement of three fish (arrows) by a dolphin (full triangle) ($\alpha = \frac{\pi}{20}, \beta = \frac{\pi}{3}$).

Next, we consider the case where a fish bounces off the boundary of the ROI following Snell's law, i.e., the angle of incidence, γ and reflection measured from the normal to the boundary of ROI at point P are equal. As Figure 33 suggests, the reflection angle β measured from the velocity vector of dolphin (center line of ROI) is computed as

$$\beta = \theta + 2\alpha - \iota, \tag{28}$$

where ι is the incidence angle measured from the positive x axis. The rest follows as the previous case, i.e., the dolphin agent's velocity can be computed using (25) or (27).

Proposition 6.3.2 *In the case of Snell's law, the reflection angle γ increases by 2δ at each bounce when $\alpha = 0$.*

Proof We know that the incidence angle of the next bounce (denoted by the superscript $^+$) is equal to the reflection angle of the previous bounce measured from the horizon, $\iota^+ = \beta + \theta$. Also, from the geometry of the problem we have $\iota^+ = \frac{3\pi}{2} + \theta^+ + \alpha + \gamma^+$ and $\beta = \frac{\pi}{2} + \alpha - \gamma$. Referring to (26) one can see that $\theta^+ = \theta + 2(\beta - \delta)$. Hence,

$\gamma^+ = \gamma + 2(\delta - \alpha)$. This concludes our proof as the reflection angle increases by 2δ when $\alpha = 0$.

As one can imagine this can cause problems when γ increases to more than $\frac{\pi}{2}$, as the fish will no longer hit the same side of the ROI and the perpetual confinement renders impossible. Figure 34 depicts the simulation of this scenario with one dolphin and two fish. To avoid this problem one can set $\alpha = \delta$ which renders γ (in turn β) constant and leads us to the following result:

Corollary 6.3.2 *In the case of Snell's law, perpetual confinement is only possible when $\alpha = \delta$.*

Proof When $\alpha = \delta$ the reflection angle $\gamma^+ = \gamma + 2(\delta - \alpha)$ stays constant which in turn results in a constant $\beta = \frac{\pi}{2} + \alpha - \gamma$.

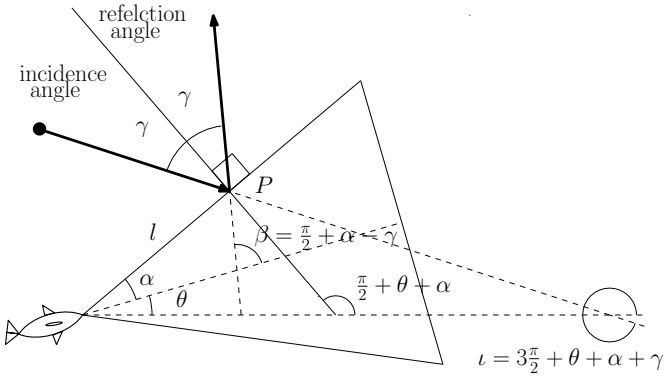


Figure 33: Geometry when reflection angle follows Snell's law.

Notice that prey with more sophisticated behavior can easily defeat the dolphin agents in their attempt of perpetual confinement (e.g., prey agents that do not maintain a constant velocity). The P^4 algorithm was designed under the assumption that the fish bounces off the ROI and the nature of this reflection is known by the dolphin. In the next section, instead of analyzing different fish behavior (and making it available to the dolphins), we identify the regions from which the fish are captured

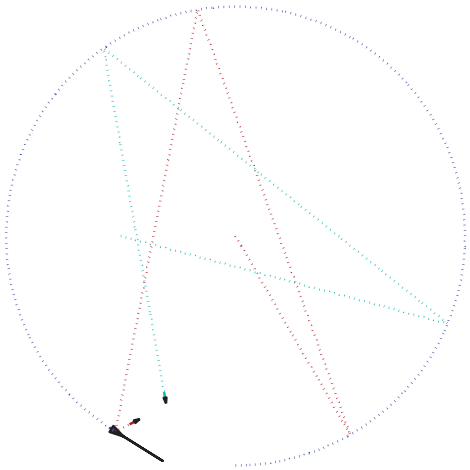


Figure 34: Simulation when reflection follows Snell's law and $\alpha = 0$ (essentially making ROI a plate). It shows that reflection angle γ is increasing by 2δ, in terms of the global coordinates, at each bounce making perpetual confinement impossible.

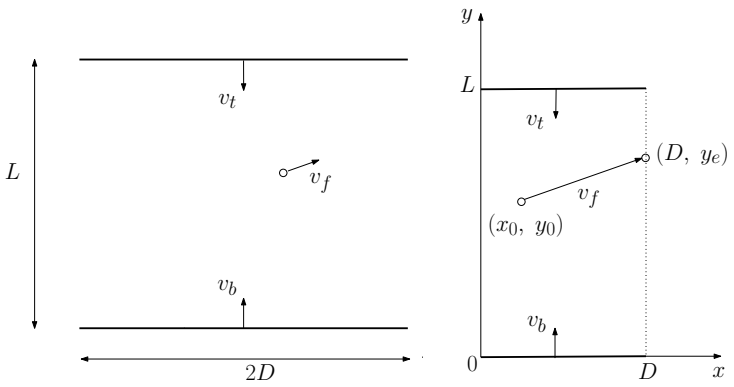
under the assumption that they are moving *optimally* to escape. Thus, we present a worst-case scenario for dolphins when we characterize these regions that guarantee capture of prey. In terms of robotic applications, this would correspond to a more pronounced non-cooperative scenario (e.g., intruder detection and capture) as opposed to the perpetual confinement scenario.

6.4 Zones Of No Escape (ZONE)

We will derive ZONE for the two variations of the wall method, fish in front and two frontal attacks, and the horizontal carousel method.

6.4.1 Wall Method: Fish in Front

In the wall method, a group of dolphins either drive the fish towards a barrier (either the shore or a stationary group of dolphins) or two groups of dolphins drive the fish towards each other. Since the dolphins move line abreast during their execution of the wall method, and with our definition of a kill zone, we can represent the dolphins in the wall method by bars, as shown in Figure 35(a).



(a) The two bars are each $2D$ in length and initially a distance L apart.

(b) The initial position of the fish is $(x_0,\ y_0)$ and the exit point is denoted by $(D,\ y_e)$.

Figure 35: Dolphins in the wall method as represented as "bars". The top and bottom bar are moving at a speed of v_t and v_b, respectively. The fish are moving with a speed of v_f at constant heading. Using a Cartesian coordinate system, only the right-half plane is analyzed due to the symmetry.

Using a Cartesian coordinate systems and exploiting the symmetry about the y-axis, we only analyze the right half plane as shown in Figure 35(b). The distance between the two bars is L and the length of each bar is $2D$. The initial position of the fish is denoted by $(x_0,\ y_0)$, where $x_0 \in [0,\ D]$ and $y_0 \in [0,\ L]$. The bottom bar is moving with a constant speed of v_b towards the top bar and in the fish in front variation of the wall method, the top bar is stationary, i.e., $v_t = 0$. To find the guaranteed region of entrapment, we assume that fish agents are making a beeline for the exit point, denoted by $(D,\ y_e)$. Therefore, the fish are moving with a constant speed of v_f at a constant heading towards this exit point. In our formulation, *optimal exit point* refers to the exit point which maximizes the meeting time between the driving bar and the fish and is denoted by $(D,\ y_e^\star)$.

The distance a fish needs to travel to escape from dolphins is given by

$$d_f = \sqrt{(D - x_0)^2 + (y_e - y_0)^2}$$

and the time it takes the fish to travel this distance is

$$\tau_f = \frac{d_f}{v_f}.$$

On the other hand, the time it takes the dolphins to reach the same point is

$$\tau_b = \frac{y_e}{v_b}.$$

To find the optimal exit point for a fish starting from the position $(x_0,\ y_0)$, we solve the following optimization problem:

$$\max_{y_e}\ \tau_b - \tau_f$$
$$s.t.\ 0 \leq y_e \leq L.$$

If we let $F(y_e) = \tau_b - \tau_f$ and set $\frac{\partial F(y_e)}{\partial y_e}(\tilde{y_e}) = 0$, then we obtain the following expression for $\tilde{y_e}$:

$$\tilde{y_e} = y_0 + \frac{v_f}{\sqrt{v_b^2 - v_f^2}}(D - x_0).$$

The cases when $v_b = v_f$ and $v_b < v_f$ are inadmissible and it turns out that for the both of these cases,

$$\arg\max_{y_e}\ F(y_e) = L.$$

As a result, we can combine these two cases as being a single case where $y_e^\star = L$.

Case 1. $v_b \leq v_f$: In this case, as mentioned before, $y_e^\star = L$ and at the optimal exit point $(D,\ y_e^\star)$, the cost function is

$$F(y_e^\star) = \frac{L}{v_b} - \frac{\sqrt{(D - x_0)^2 + (L - y_0)^2}}{v_f}.$$

By setting the cost function to 0 we have the following expression

$$(D - x_0)^2 + (L - y_0)^2 = \left(L\frac{v_f}{v_b}\right)^2. \tag{29}$$

Equation (29) is a circle centered at the point $(D,\ L)$ with radius $L\frac{v_f}{v_b}$ and the equation represents the initial positions of the fish from which they arrive at the optimal exit point at the same time as the driving bar. If we assume that in this situation, the dolphins are able to catch the fish, then these set of initial positions form the boundary of the ZONE, as shown in Figure 36. The ZONE for the right-half plane, are the points on and to the left of the ZONE boundary (by Lemma 6.4.3).

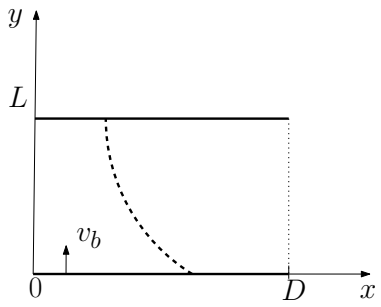


Figure 36: The boundary of the ZONE (dash) represents the initial positions from which the bottom bar and fish arrive at the optimal exit point at the same time for the case $v_b \leq v_f$.

Lemma 6.4.1 ZONE $= \emptyset$ *if* $L^2 + D^2 \leq L\frac{v_f}{v_b}$.

Lemma 6.4.1 mathematically states that there exists a condition on the length of the driving bars, the initial distance between the bars, and the predator and prey speeds, for which there are no regions that guarantee capture of prey. Intuitively, for the wall method with given predator and prey speeds, it makes sense that there are fewer regions that can be classified as ZONE when dolphins drive the fish for longer distances (larger L), or when the length of the driving bars is shorter (smaller D).

Case 2. $v_b > v_f$: In this case, $y_e^\star = \tilde{y_e}$ if $\tilde{y_e} \leq L$ and by setting the cost function to 0 at the optimal exit point, as we did in the previous case, we obtain a degenerate hyperbola as the shape of the critical initial positions from which the fish and the moving bar arrive at the exit point at the same time. However, as we are

only interested in the right-half plane, the boundary of the ZONE is actually a line with the following equation:

$$y_0 = (D - x_0)\sqrt{\frac{v_b^2 - v_f^2}{v_f^2}}. \tag{30}$$

But when $\tilde{y_e} > L$, the optimal exit point is $(D,\ L)$ and we again have a circle centered at $(D,\ L)$ with radius $L\frac{v_f}{v_b}$. The ZONE for the right-half plane are all the points on and to the left of (by Lemma 6.4.3) the shape shown in Figure 37.

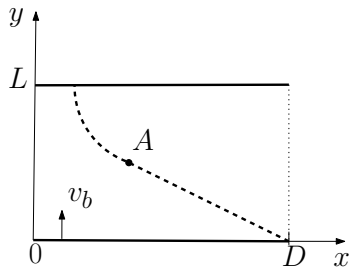


Figure 37: The boundary of the ZONE (dash) when $v_b > v_f$. At the point A, $\tilde{y_e} = L$. If $\tilde{y_e} \leq L$, the boundary is given by (30) and if $\tilde{y_e} > L$ the boundary is given by (29).

Lemma 6.4.2 *The boundary of the* ZONE *shown in Figure 37 is continuous.*

Proof Let A denote the point on the line with (30) where $\tilde{y_e} = L$. The distance between the point $(D,\ L)$ and the point A is $L\frac{v_f}{v_b}$, which is the radius of the circular part of the ZONE boundary.

Lemma 6.4.3 *If a fish cannot escape from an initial position* $(x_0,\ y_0)$*, then it cannot escape from the initial position* $(x_0',\ y_0)\ \forall\ 0 \leq x_0' < x_0 \leq D$.

Proof Consider fish 1 with initial position $(x_0^1,\ y_0)$ and fish 2 with initial position $(x_0^1 - \Delta,\ y_0)$, where $\Delta > 0$. We need to show that the distance fish 1 needs to travel to escape, d_1, is always smaller than d_2, the distance fish 2 needs to travel to escape.

Table 3: ZONE for fish in front.

Conditions	ZONE
$v_f \geq v_b$	$\left\{(x_0,\ y_0) \mid (D-x_0)^2 + (L-y_0)^2 \geq \left(L\frac{v_f}{v_b}\right)^2\right\}$
$v_f < v_b\ \wedge\ \tilde{y_e} > L$	$\left\{(x_0,\ y_0) \mid (D-x_0)^2 + (L-y_0)^2 \geq \left(L\frac{v_f}{v_b}\right)^2\right\}$
$v_f < v_b\ \wedge\ \tilde{y_e} \leq L$	$\left\{(x_0,\ y_0) \mid y_0 \leq (D-x_0)\sqrt{\frac{v_b^2-v_f^2}{v_f^2}}\right\}$

For $v_f \geq v_b$: The optimal exit point is $(D,\ L)$ for both fish. In this case,

$$d_1 = \sqrt{(D-x_0^1)^2 + (L-y_0)^2}$$

and

$$d_2 = \sqrt{(D-x_0^1+\Delta)^2 + (L-y_0)^2} > d_1.$$

For $v_f < v_b$: Let $\tilde{y_{e,1}}$ and $\tilde{y_{e,2}}$ denote $\tilde{y_e}$ evaluated for fish 1 and 2, respectively. If $\tilde{y_{e,1}} > L$, then $\tilde{y_{e,2}} > L$; thus both fish have the same optimal exit point $(D,\ L)$ again. If $\tilde{y_{e,1}} \leq L$, fish 1 needs to travel

$$d_1 = (D-x_0^1)\frac{v_b^2}{v_b^2-v_f^2}.$$

If $\tilde{y_{e,2}} \leq L$, fish 2 needs to travel

$$d_{2,1} = (D-x_0^1+\Delta)\frac{v_b^2}{v_b^2-v_f^2} > d_1.$$

If $\tilde{y_{e,2}} > L$, fish 2 needs to travel

$$d_{2,2} = \sqrt{(D-x_0^1+\Delta)^2 + (L-y_0)^2} \geq d_{2,1} > d_1.$$

The ZONE (restricted to the right-half plane) for all cases of the fish in front method are presented in Table 3.

6.4.2 Wall Method: Two Frontal Attacks

In this variation of the wall method, the top bar is no longer stationary and moves towards the bottom bar with a constant speed of v_t. We solve the problem of two moving bars by solving two separate problems, each with one moving bar and one stationary bar. The two moving bars will meet at a certain time, and the position of the bars at this time, $y = L^\star$ will be viewed as a virtual stationary bar, where $L^\star = \frac{v_b}{v_b+v_t}L$.

When the fish are initially on or below the virtual bar, we only consider the problem where the bottom bar moving towards the virtual stationary bar. For the fish initially above this virtual bar, the optimization problem only considers the top bar driving the fish towards the virtual bar. With this formulation, we have two one-moving bar problems and all we need to do is follow the steps prescribed in the fish in front method.

When $y_0 \leq L^\star$, the problem is similar to the fish in front problem (solved in the last section), where the bottom bar is moving towards a stationary top bar; except here, the bottom bar and the stationary bar (virtual bar), are a distance $L^\star$ apart. Thus, we can replace L by $L^\star$ in the expression for the ZONE obtained in the fish in front method to express the ZONE when $y_0 \leq L^\star$ in the two frontal attacks method.

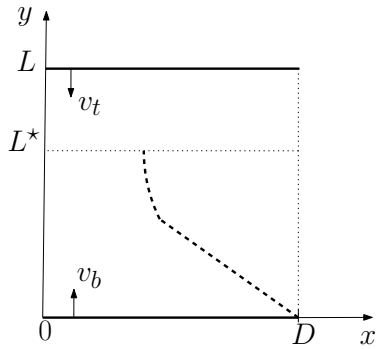


Figure 38: The two moving bars meet at the position $y = L^\star$. The boundary of the ZONE (dash) is shown $\forall\ y_0 \leq L^\star$ and $v_b > v_f$.

When $y_0 > L^\star$, we only consider the top and virtual bar. The time it takes the top bar to reach the exit point $(D,\ y_e)$ is given by

$$\tau_t = \frac{L - y_e}{v_t}.$$

We have the following optimization problem:

$$\max_{y_e} \tau_t - \tau_f$$
$$s.t.\ L^\star \leq y_e \leq L.$$

This cost function is labeled as $F_t(y_e)$ and by setting $\frac{\partial F_t(y_e)}{\partial y_e}(\hat{y_e}) = 0$, we obtain the following expression for $\hat{y_e}$:

$$\hat{y_e} = y_0 + \frac{v_f}{\sqrt{v_t^2 - v_f^2}}(D - x_0).$$

When $v_f \geq v_t$, $y_e^\star = L^\star$ and the boundary of the ZONE is a circle centered at $(D,\ L^\star)$ with radius $(L - L^\star)\frac{v_f}{v_t}$. When $v_f < v_t$ and $\hat{y_e} > L$ the boundary of the ZONE is also a circle centered at $(D,\ L^\star)$ with radius $(L - L^\star)\frac{v_f}{v_t}$. When $v_f < v_t$ and $\hat{y_e} \leq L$, the boundary of the ZONE is the line

$$y_0 = L - (D - x_0)\frac{\sqrt{v_t^2 - v_f^2}}{v_f}.$$

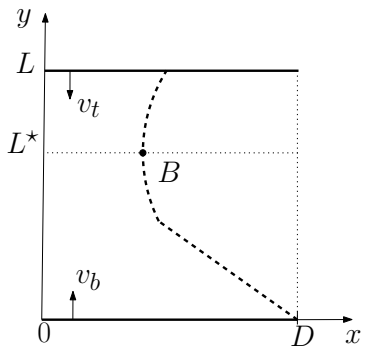


Figure 39: The two moving bars meet at the position $y = L^\star$. The boundary of the ZONE (dash) for the two moving bars when $v_t = v_f$ and $v_f < v_b$.

The ZONE (restricted to the right-half plane) is presented in Table 4 when $y_0 \leq L^\star$; in Table 5 when $y_0 > L^\star$.

Table 4: ZONE for two frontal attacks $\forall\ y_0 \leq L^\star$.

Conditions	ZONE
$v_f \geq v_b$	$\left\{ (x_0,\ y_0) \mid (D-x_0)^2 + (L^\star - y_0)^2 \geq \left(L\frac{v_f}{v_b+v_t}\right)^2 \right\}$
$v_f < v_b\ \wedge\ \tilde{y_e} > L^\star$	$\left\{ (x_0,\ y_0) \mid (D-x_0)^2 + (L^\star - y_0)^2 \geq \left(L\frac{v_f}{v_b+v_t}\right)^2 \right\}$
$v_f < v_b\ \wedge\ \tilde{y_e} \leq L^\star$	$\left\{ (x_0,\ y_0) \mid y_0 \leq (D-x_0)\sqrt{\frac{v_b^2-v_f^2}{v_f^2}} \right\}$

Table 5: ZONE for two frontal attacks $\forall\ y_0 > L^\star$.

Conditions	ZONE
$v_f \geq v_t$	$\left\{ (x_0,\ y_0) \mid (D-x_0)^2 + (L^\star - y_0)^2 \geq \left(L\frac{v_f}{v_b+v_t}\right)^2 \right\}$
$v_f < v_t\ \wedge\ \hat{y_e} > L^\star$	$\left\{ (x_0,\ y_0) \mid (D-x_0)^2 + (L^\star - y_0)^2 \geq \left(L\frac{v_f}{v_b+v_t}\right)^2 \right\}$
$v_f < v_t\ \wedge\ \hat{y_e} \leq L^\star$	$\left\{ (x_0,\ y_0) \mid y_0 \geq L - (D-x_0)\frac{\sqrt{v_t^2-v_f^2}}{v_f} \right\}$

Lemma 6.4.4 *In the one-moving bar case, if the boundary of the* ZONE *intersects a stationary bar, it does so as a circle.*

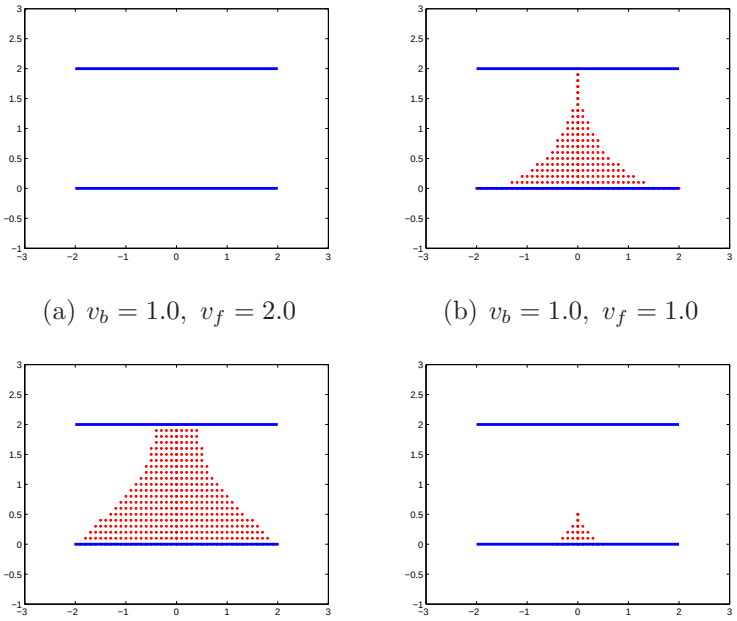
Proof Let us consider the bottom bar moving towards a stationary bar, a distance L away. (The opposite case is not shown because the proof is similar). When $v_f \geq v_b$, the boundary of the ZONE is a circle with radius $L\frac{v_f}{v_b}$. When $v_f < v_b$, the boundary of the ZONE is a circle with radius $L\frac{v_f}{v_b}$ if $\tilde{y_e} = y_0 + \frac{v_f}{\sqrt{v_b^2-v_f^2}} > L$. At the stationary bar, $y_0 = L$ and thus, $\tilde{y_e} > L$ is always true. Therefore, if the boundary of the ZONE intersects the stationary bar, it does so as a circle.

Lemma 6.4.5 *For two moving-bars, if the boundary of the* ZONE *intersects the virtual bar, then the boundary is continuous.*

Proof By Lemma 6.4.4, the ZONE boundary derived by each moving bar intersect the virtual stationary bar as a circle (see Figure 39). Both circles are centered at

$(D,\ L^\star)$ and the circle associated with the bottom bar has a radius $L^\star\frac{v_f}{v_b}$ and the circle associated with the top bar has a radius $(L-L^\star)\frac{v_f}{v_t}$. Since $L^\star = L\frac{v_b}{v_b+v_t}$, it turns out that the two radii are equal.

Simulations of ZONE for the fish in front and the two frontal attacks methods are shown in Figure 40 and Figure 41, respectively.



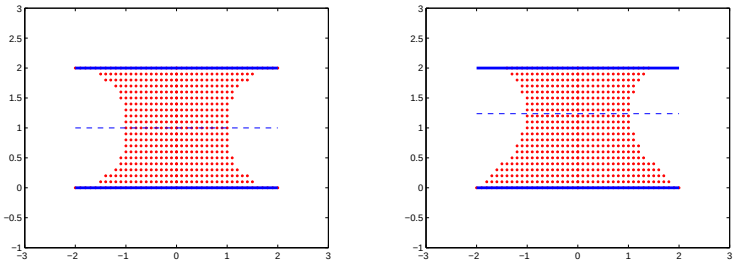
(a) $v_b = 1.0,\ v_f = 2.0$ (b) $v_b = 1.0,\ v_f = 1.0$

(c) $v_b = 1.5,\ v_f = 1.0$ (d) $v_b = 1.0,\ v_f = 1.2$

Figure 40: The dots represent the ZONE for the fish in front method. The bottom bar is driving towards the stationary top bar. The initial position of the bars are shown as bold lines.

6.4.3 Horizontal Carousel

In this method, as mentioned before, the dolphins first encircle the fish and tighten this encirclement to restrict the movement of the fish. Let us assume that a group of N dolphins employs the vertical carousel method to eat the fish after constricting them inside a circle by using the horizontal carousel technique. The dolphins are assumed to be equally spaced on a circle and each dolphin is assumed to be traveling with a constant speed of v_b towards the centroid of the fish, C.



(a) $v_b = 1.0,\ v_t = 1.0$ (b) $v_b = 1.3,\ v_t = 0.8$

Figure 41: The dots represent the ZONE for the two frontal attacks method when $v_f = 1$. The bold lines represent the initial position of the bars and are identical to those of figure 40. The dashed line shows the position where the bars eventually meet.

Definition 6.4.1 *A hole is the shortest distance between the kill zones of two neighboring dolphins.*

In Figure 42, the initial length of the holes is $2K_0$. If two kill zones first overlap at $t = t^\star$, then there are no more holes created by the dolphins $\forall\ t \geq t^\star$. This is important for the fish agents since they will need to exploit the holes in order to escape the dolphins. If a fish is surrounded by dolphins and there are no holes left, then that fish is guaranteed to be eaten.

Due to the symmetric nature of the problem, we only need to analyze one dolphin driving towards the centroid of the fish. Furthermore, we can also exploit the symmetric nature of the kill zone and only analyze one half of the kill zone driving towards the fish centroid as shown in Figure 43(a). As mentioned earlier, the fish need to exploit the holes in the dolphin arrangement; consequently, to escape, the fish initially positioned in the light shaded area must move to dark shaded area (defined by the holes) to escape form the dolphins (see Figure 43(a)).

If the length of the kill zone is $2D$, then we again have a moving bar of length D and the exit point for the fish is again $(D,\ y_e)$, where $y_e \in [0,\ L]$ as shown in Figure 43(b). The set up of the problem is different from the "fish in front" set up in

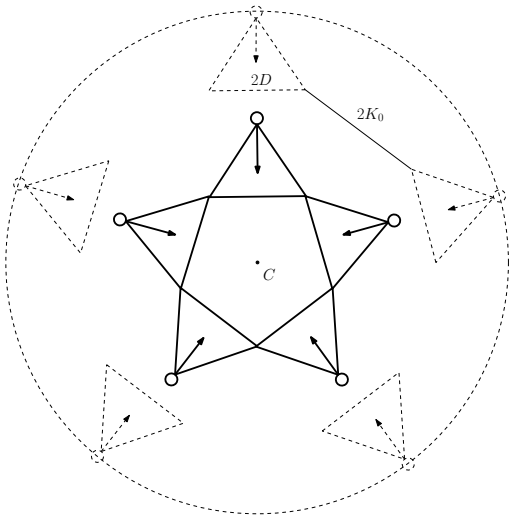


Figure 42: Dolphins (open circles) charge through the school of fish (represented by its center of mass, C) in unison. The length of the holes are initially $2K_0$. If two kill zones first overlap at $t = t^\star$, then the dolphin arrangements have no holes $\forall\ t \geq t^\star$.

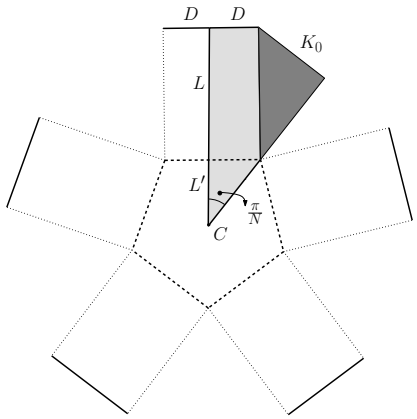
only two ways. First, there is a larger set of possible initial positions for the fish and this depends on the length L' as shown in Figure 43(b). L' is the shortest distance between the centroid and the killzone at time $t^\star$ and is given by

$$L' = \frac{D}{tan\frac{\pi}{N}}.$$

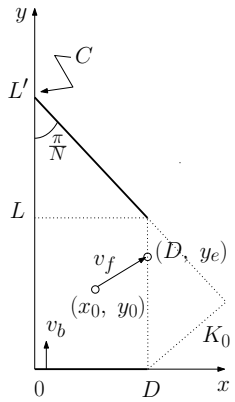
The second difference is that the distance L is pre-specified in the "fish in front" case, while in the horizontal carousel case, this distance depends on both the number of agents participating in the hunt and the initial length of a hole. For the horizontal carousel, L is given by

$$L = \frac{K_0}{sin\frac{\pi}{N}}. \tag{31}$$

Increasing the set of possible initial positions for the fish does not affect the ZONE found in the "fish in front" case and by using the definition of L from (31) into the ZONE derived by the "fish in front" case, we can find ZONE for the horizontal carousel case as shown in Figure 44(a), where $v_b \leq v_f$; in Figure 44(b), where $v_b > v_f$. The ZONE for five dolphins using the horizontal carousel method is shown Figure 45.



(a) Fish initially in the light shaded area must move to the dark shaded area to escape. The dashed lines are the positions of the bars when two kill zones first overlap.

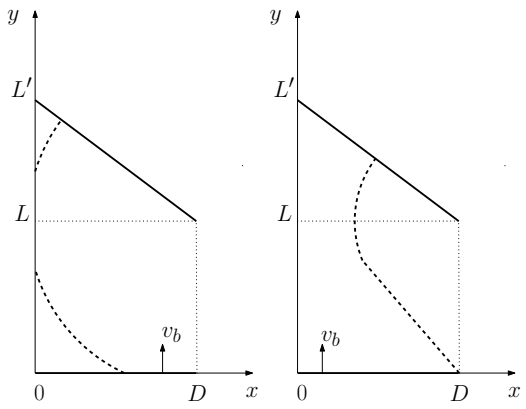


(b) Due to the symmetry, one half of a kill zone is analyzed.

Figure 43: $N = 5$ dolphins are positioned to drive towards the centroid of fish (C). The initial length of a hole is $2K_0$. The kill zones are $2D$ in length. L' is the shortest distance between the centroid and the kill zone at time $t^\star$. L is the distance dolphins travel before two kill zones first overlap.

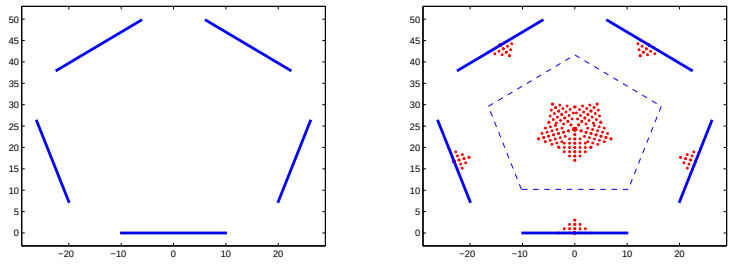
6.5 Conclusion

In this chapter, confinement strategies were developed for a multi-agent system based on the foraging techniques of Bottlenose dolphins. We started with the simple case where the evasive maneuvers of the fish were known by the dolphin agents. Conditions were derived for a single dolphin to perpetually confine a school of fish. Next, we assumed that fish behaved optimally in their attempt to escape and subsequently, we characterized the regions from which they were guaranteed to be captured.

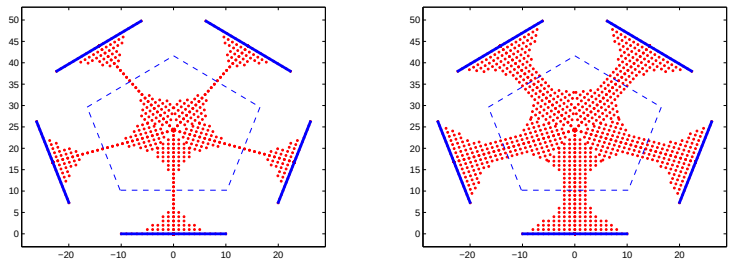


(a) $v_b \leq v_f$. (b) $v_b > v_f$.

Figure 44: Boundary of the ZONE (dash) for one-half of a kill zone during horizontal carousel.



(a) $v_b = 1.0,\ v_f = 2.0$ (b) $v_b = 0.8,\ v_f = 1.0$



(c) $v_b = 1.0,\ v_f = 1.0$ (d) $v_b = 1.0,\ v_f = 0.8$

Figure 45: The dots represent the ZONE for the horizontal carousel method. The bold lines represent the initial position of the bars and the dashed lines represent the position of the bars when the kill zones first overlap.

CHAPTER VII

GEOMETRIC FORAGING STRATEGIES

Social animals often form a predator front to charge through an aggregation of prey. Inspired by this observation, we model the foraging multi-robot front as a curve moving through a prey density and optimize the shape of the front.

7.1 Introduction

Bottlenose dolphins and African lions are examples of biological systems that arrange themselves in specific formations to create predator fronts that move together, in unison, towards the collection of prey.

Variations exist in the shape of the charging front; in fact, dolphin fronts are different from those of lions and this difference can be attributed to the nature of their feeding strategies, e.g., [23, 75]. In the first part of this chapter, we recover these differences by optimizing over the shape of the front for a given feeding strategy. More specifically, we simulate a set quadratic curves through the aggregation of prey, the movement of which is specified e.g., by a reaction-diffusion process.

In the second part of the chapter, a curve flow algorithm is developed that maximizes the total energy intake, i.e. the total amount of prey swept by the front. The algorithm is based on curve evolution techniques, which are widely used in the field of image processing, e.g. see [11, 53, 91]. Active contours for image segmentation evolve an initial curve under a cost function to detect objects. One common approach is to model the initial curve as a level set and define the optimality condition based on the speed of the curve, e.g. [11, 91]. Here, an arc length parametrized curve is evolved according to a gradient ascent based deformation algorithm.

One potential application for this work is the clean up of oil spills. Until now,

unmanned vehicles have been deployed in the spill site to collect data on ocean properties and survey the extent of damage. However, we propose to utilize a multi-robot system for an efficient clean up task of spilled oil. Figure 46 shows how a group of robots coordinate to drive a flexible suction boom towards a spill site. Using the proposed curve flow algorithm, we can optimize the shape of this boom for an efficient cleanup of the oil spill as a function of the oil dynamics.

The remainder of the paper is organized as follows: in Section 7.2, we find the most efficient curve for two distinct feeding strategies - one that is inspired by African lions and other inspired by Bottlenose dolphins - by simulating a set of quadratic curves through the aggregation of prey. Discussions, based on the results of Section 7.2 are presented in Section 7.3. In Section 7.4, we optimize the shape of the curve by developing a curve deformation algorithm. The cost function used in Section 7.4 is modified in Section 7.5 to develop a alternative curve deformation algorithm. Finally, Section 7.6 presents the conclusions of this chapter.

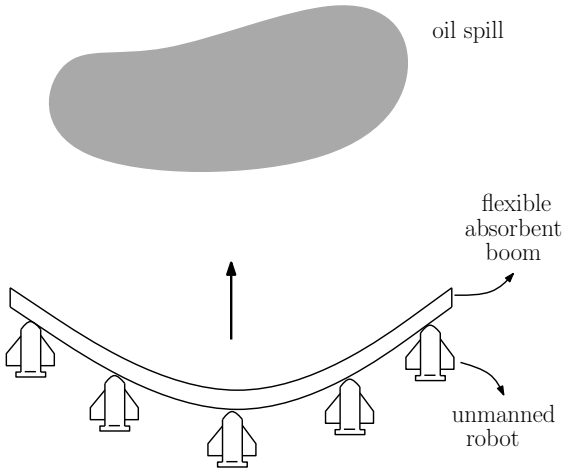
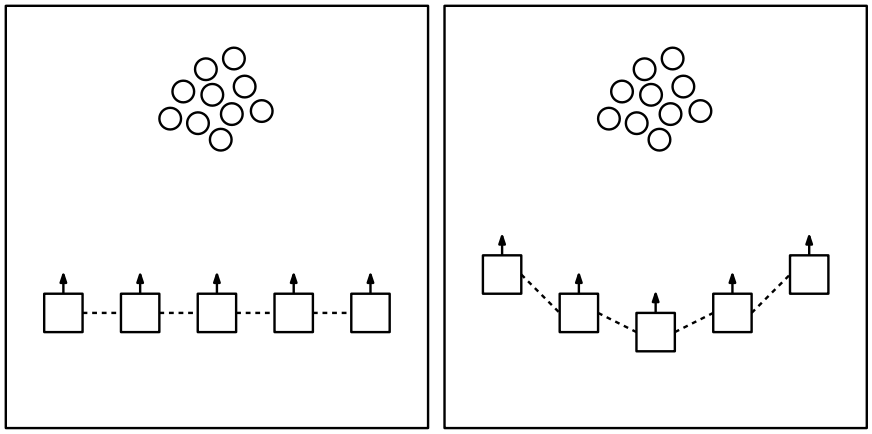


Figure 46: A group of unmanned vehicles are driving a flexible, absorbent boom towards an oil spill. Optimizing the shape of the boom, to remove the most amount of oil, is a possible application of the proposed curve flow algorithm.



(a) Dolphins driving in a "line abreast" formation.

(b) Lions driving in a "catcher's mitt" formation.

Figure 47: The arrangement of foragers in the predator front is shown for dolphins and lions. Predators (squares) are driving towards (arrows) the direction of their prey (circles).

7.2 Predator Fronts as Quadratic Curves

As described in Chapter 2, in the wall method, dolphins arrange themselves next to each other to create a front that charges through a school of fish, as shown in Figure 47(a). The shape of the front is described in [75] as being "line abreast." African lions implement a prey capturing technique quite similar to the wall method. When hunting medium-sized prey like zebra, a single lion has a low success rate (about 17%) [23] and as a result, lions revert to group hunting. Female lions are usually in charge of foraging and while charging towards their prey, lionesses in the "wing" positions cause the prey to drive towards the lionesses waiting in the "center" positions [73]. The resulting predator front that drives towards the prey is therefore *U-shaped*, often described as a "catcher's mitt" [23] as shown in Figure 47(b).

A zebra typically weighs around 250 kg and although it is brought down by a singe lion, others in the group claim their share and earn a free lunch. On the other hand, each member of the dolphin foraging group only eats what it can catch. We refer to these two distinct feeding strategies as the *free lunch* (lion-inspired) and the *no-free*

lunch case (dolphin-inspired). In this section, for a given feeding strategy, we obtain the most efficient predator front through the solution to an optimization problem, where the free lunch curve maximizes the total energy intake and the no-free lunch curve maximizes the energy of the agent that feeds the least.

We specify a candidate set of curves *apriori*, and each curve from this candidate set is swept through the aggregation of prey. More specifically, we only consider quadratic curves, where each candidate curve has the same arc length. The constant arc length requirement is placed from an engineering design perspective. If our multi-agent system consists of M agents, each with a limited communication range, it is desirable to restrict inter-agent distances to remain within this range so that each agent remains in constant communication with its neighboring agents. Since our predator agents create a front and charge towards the prey while maintaining the shape of the front, we simply require all of the candidate curves to be of the same arc length, ξ; thus if we have M agents, the inter-agent arc length remains $\xi/(M-1)$.

Consider the curve of the form $y = ax^2 + K$ and arc length ξ. In fact, due to the constant length assumption, the coefficient a depends on K. Furthermore, if the endpoint of the curves are $(-x_m, 0)$ and $(x_m, 0)$, where $x_m > 0$, then the K-parametrized curve, $Q(K)$, is given by

$$y = \frac{-K}{x_m^2}x^2 + K, \tag{32}$$

where, x_m is given by solving the following equation:

$$\begin{aligned} \xi &= \int_{-x_m}^{x_m} \sqrt{1 + \frac{dy}{dx}}\, dx \\ &= \int_{-x_m}^{x_m} \sqrt{1 + \frac{-2K}{x_m^2}x}\, dx. \end{aligned} \tag{33}$$

This formulation gives us the shape of the candidate curve for each value of K. If the candidate curves under consideration include all curves between $Q(K_i)$ and $Q(K_f)$, then we can denote the set of candidate curves as $\mathcal{C} = \{Q(K) \mid K \in [K_i, K_f]\}$, as shown in Figure 48.

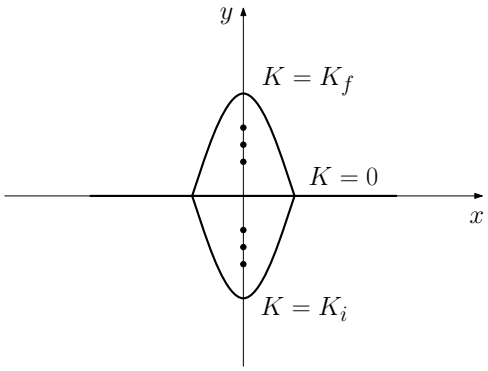


Figure 48: The curves are parameterized by K and the candidate curves being swept through the aggregation of prey include all the curves between $K = K_i$ and $K = K_f$.

Each K-parameterized curve charges towards the prey aggregation and if we assume, without loss of generality, that the front charges in the direction of increasing y, from $t = 0$ to $t = t_f$ with a speed v_c, then at time t, the curve $Q(k)$ is given by $y = \frac{-K}{x_m^2}x^2 + K + v_c t$.

7.2.1 Prey Aggregation as a Density Function

The $2D$ prey density is denoted by $u(x, y, t)$, where $u : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}$ and (x, y) represents Cartesian coordinates. We consider two types of processes to define prey movement, which in turn describes our predator-prey interaction, one is a simple diffusion and the other is a reaction-diffusion. Representing prey as a density function and using reaction-diffusion equations to model the spatio-temporal profile of prey is formally known as the "population framework" to model prey [54]. We use this approach, as opposed to an agent-based model of prey, such as the one presented in [42], as we are interested in the collective movement of prey, whereas the agent-based approach requires us to define control laws for the movement of individual prey-like agents. The details of our movement laws are discussed in more detail below.

1) Diffusion: The first type of prey movement is a diffusion given by

$$\frac{\partial u(x, y, t)}{\partial t} = v_0 \left(\frac{\partial^2 u(x, y, t)}{\partial x^2} + \frac{\partial^2 u(x, y, t)}{\partial y^2} \right), \tag{34}$$

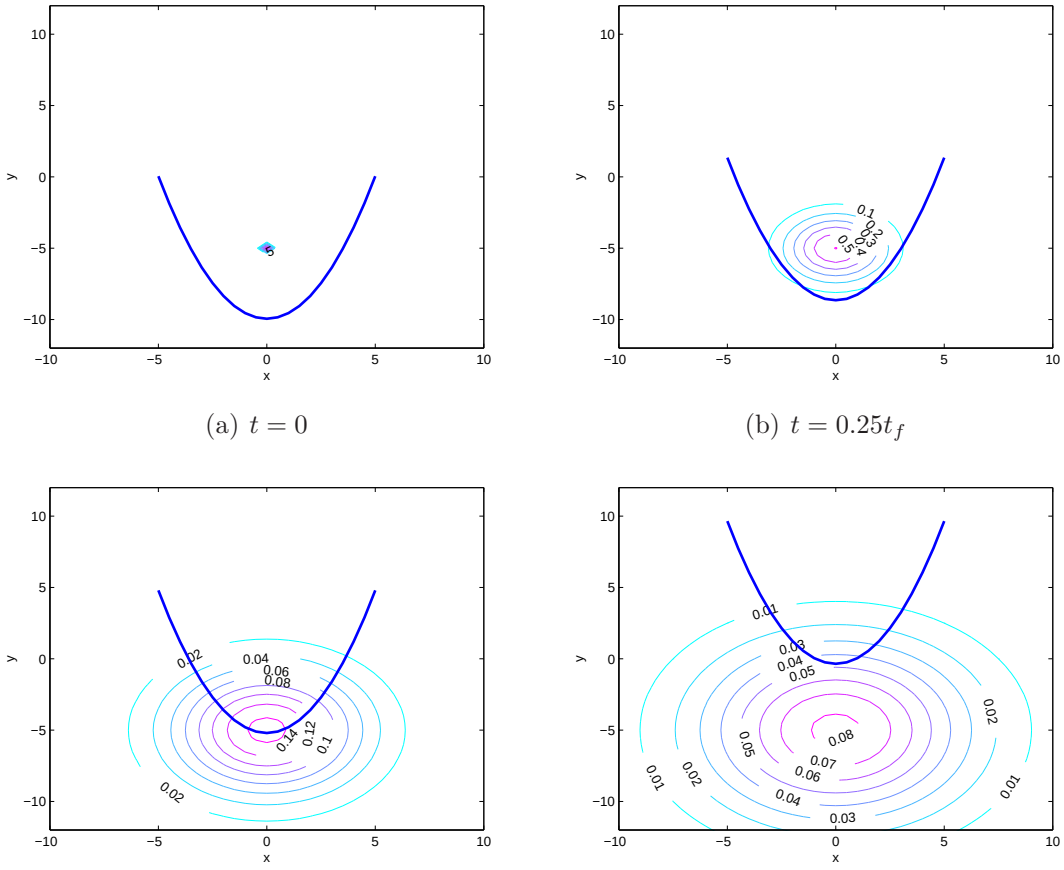
where, $v_0 \in \mathbb{R}_+$ is the thermal diffusivity. As a movement law for prey, (34) models the case where there is no predator-prey interaction. The prey diffuses from its initial density, $u(x, y, 0)$, at a "speed" of v_0, regardless of the location of the predator front. The diffusion of the prey is shown as contour levels in Figure 49.

Besides the cleanup up oil spills, another application of this work would be securing littoral regions using teams of autonomous vehicles as described by the US Navy Sea shield mission [1]. Threats in the littoral regions usually consist of mines, submarines, and unmanned underwater vehicles (UUVs), which must be neutralized before other teams can land on shore. Equation (34) captures the movement of objects like floating mines, as opposed to advanced mines (for details see [80]). The reaction-diffusion process described next, models the movement of more sophisticated threats, such as UUVs that react to the location of the foraging robots.

2) Reaction-Diffusion: A reaction-diffusion process is a more natural representation of the prey movement than a simple diffusion process (as the one used in the previous subsection) since it incorporates the prey response to a predator charge. In general, a reaction-diffusion process models the changes in a substance under: 1) reaction - the influence of another substance and 2) diffusion - the spatial distribution. There are numerous mathematical models of a reaction–diffusion process and the one we use is known as the FitzHugh-Nagumo model. Because of its simplicity, this model is widely used in the field of mathematical biology to describe the firing of neurons and the propagation of nerve action potentials under the excitation of ion movement across a membrane [63]. We tailor the system of partial differential equations used to describe the FitzHugh-Nagumo model in [63] to model the propagation of prey under the excitement of the predator front as follows:

$$\frac{\partial u}{\partial t} = v(q(x, y)) \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \sigma q(x, y), \tag{35}$$

where $\sigma \in \mathbb{R}_+$ and the diffusion coefficient, v, now depends on the predator location,



(a) $t = 0$ (b) $t = 0.25t_f$

(c) $t = 0.5t_f$ (d) $t = t_f$

Figure 49: The curve $Q(-10)$ is sweeping through the prey (represented as contour levels), which are diffusing according to (34).

$q(x, y)$. For a curve $Q(K)$, we define the predator location as follows:

$$q(x,y) = \begin{cases} 1 & \text{if } (x,y) \text{ on } Q(K) \\ 0 & \text{otherwise} \end{cases}, \ i \in \{1, \ldots, M\}. \tag{36}$$

The diffusion coefficient is modeled as

$$v(q) = \begin{cases} v_0 + \lambda & \text{if } p = 1 \\ v_0 & \text{otherwise} \end{cases} \tag{37}$$

where $\lambda \in \mathbb{R}_+$. Such a formulation for the thermal diffusivity captures the idea of the prey being "scared" in the presence of predators. For a location (x, y), when $q = 0$ (i.e., there are no predators present in that location), the prey-flock diffuses according to the nominal "speed" of v_0; but when $q = 1$, they diffuse faster at a speed of $v_0 + \lambda$. We also capture the idea of prey being "eaten" with the $-\sigma q$ term.

Next, based on our models of the predator front and the PDE-based models of prey movement, we characterize the optimal charging front in both the free lunch and no-free-lunch cases.

7.2.2 Front Design

The most efficient charging front for a given feeding strategy is obtained through the solution to an optimization problem. From the set of candidate curves, $\mathcal{C}$, the free-lunch curve maximizes the total energy intake of the foraging group and the no-free lunch curve maximizes the energy intake of the agent that feeds the least (for details about this motivation, please see [34]). But before we begin, we need to define the energy intake for an individual agent. If the position of agent i is denoted as $Q_i(K)$ and we let agent i "eat" $u(Q_i(K), t)$ amount of prey at time t, then the total amount of energy consumed, i.e. prey captured, by agent i is given by

$$E_i = \int_0^{t_f} u(Q_i(K), t) \, dt, \ i \in \{1, \ldots, M\}. \tag{38}$$

With this definition of individual energy intake, we now characterize the optimal charging curves for the free lunch and the no-free lunch case.

Lions earn a free lunch by eating the prey caught by the more skilled hunters that take positions in the center of the front. In this case, the goal is to capture as much prey as possible for a group feasting to take place. Thus, since the total energy, E, is given by

$$E = \sum_{i=1}^{M} E_i, \tag{39}$$

the free lunch curve is the curve $Q(K^\star)$, where

$$K^\star = \arg\max_K \ E \tag{40}$$

As opposed to lions, each dolphin in the foraging group must contribute in the hunt and as a result, in our formulation, the no-free lunch curve is the curve that maximizes the total energy intake of the agent that feeds the least. Let the energy of the agent that feeds the least be denoted as E', then

$$E' = \min_i \ E_i, \tag{41}$$

where $i \in \{1, \ldots, M\}$. The no-free lunch curve is the curve $Q(K')$, where

$$K' = \arg\max_K \ E' \tag{42}$$

7.2.3 Simulations

The results of our model are shown in Figure 50. The foraging area is represented as a $2D$ mesh, where $x_{min} = -30$, $x_{max} = 30$, $y_{min} = -30$, $y_{max} = 30$, and the mesh spacing is $\Delta x = \Delta y = 0.5$. The curves are swept from $t_i = 0$ to $t_f = 20$, with a time step of $\Delta t = 0.005$. The candidate curves are the curves between $K_i = -10$ and $K_f = 10$, with a step of $\Delta K = 0.5$ and arc length $L = 23$. We use 21 predator agents, thus curves are drawn using $M = 21$ equally-spaced data points.

Three distinct initial prey densities are considered. Each initial density is a ball of radius 4 units and they differ in the location of their center (denoted as 'x'). In Figures 50(a) and 50(b), the center is located at $(0, -5)$; in Figures 50(c) and 50(d),

the center is located at $(0, 0)$; and in Figures 50(e) and 50(f), the center is located at $(0, 10)$. For each position, we simulate the prey movement as a diffusion process in Figures 50(a), 50(c), and 50(e); as a reaction-diffusion process in Figures 50(b), 50(d), and 50(f). The diffusion process parameters are $v_0 = 0.5$, $\lambda = v_0$, and $\sigma = 10$.

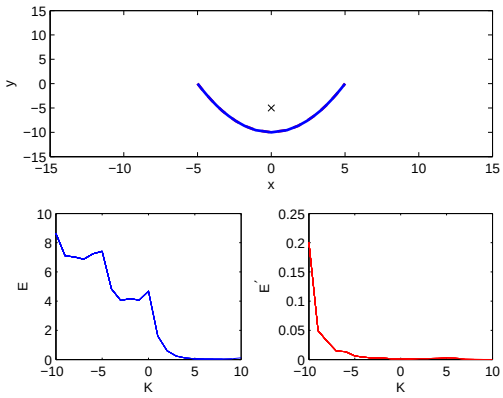
There are two curves displayed for each prey center of density and prey movement rule pair: the free lunch curve (solid line), which maximizes E, and the no-free lunch curve (dashed line), which maximizes E'.

7.3 Discussions

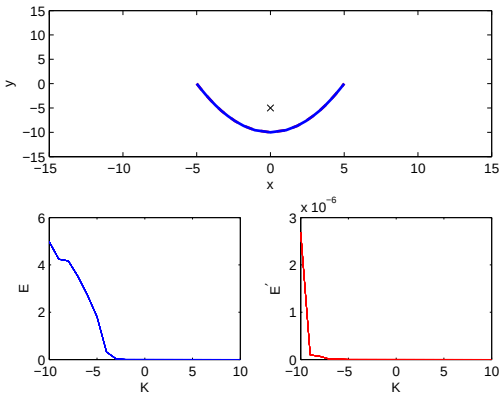
Although we produced simple biological models of charging predator fronts, we observe that our simulations render strong resemblances to actual lion fronts (i.e., the free lunch case). In Figures 50(a), 50(b), and 50(d), we notice that our lion-inspired fronts look like the catcher's mitt shape described in [23] - with agents in the wing and center positions.

In the diffusion cases of Figures 50(c) and 50(e), the lion-inspired fronts are not U-shaped. This makes sense because in these cases, as opposed to the diffusion case of Figure 50(a), the candidate fronts sweep through more cells with a prey density value of 0. Thus, in these cases, to maximize E, the optimal curves are those that tend to "hug" the prey center and capture the most available prey at the start of the sweep.

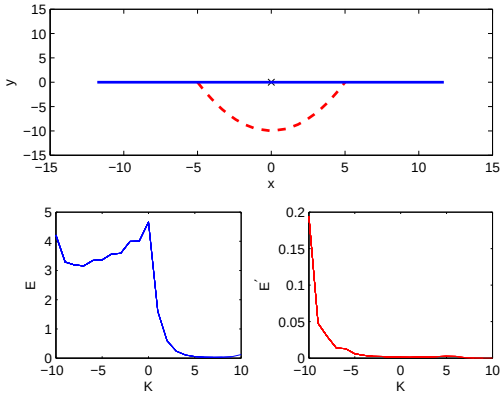
For the prey centered at $(0, -5)$, the lion-inspired fronts are the same for both the diffusion (Figure 50(a)) and the reaction-diffusion case (Figure 50(b)). However, we obtain different shapes for the diffusion and reaction-diffusion cases when the prey is centered at $(0, 0)$ and $(0, 10)$. As we mentioned earlier, in Figures 50(c) and 50(e), the optimal fronts are the ones that pass through the center of the prey at the beginning of the sweep. On the other hand, since our reaction-diffusion process models prey being scared, it turns out that the optimal thing to do is no longer to



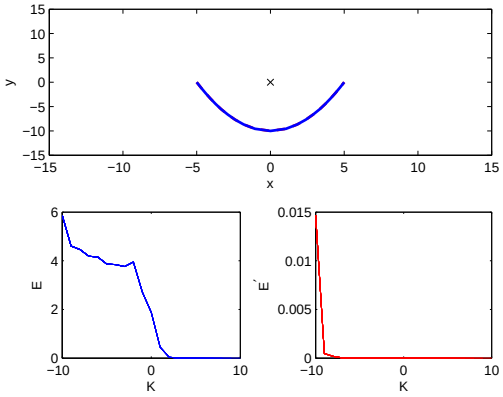
(a) Prey at (0,-5), $K^\star = -10$, $K' = -10$



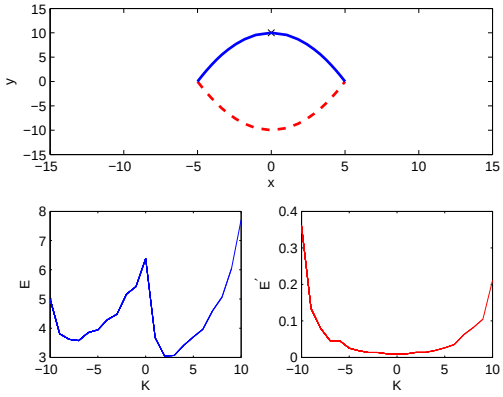
(b) Prey at (0,-5), $K^\star = -10$, $K' = -10$



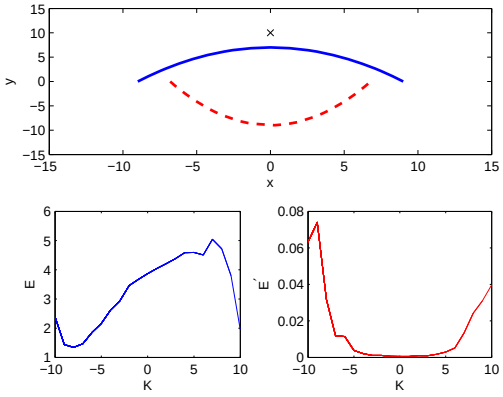
(c) Prey at (0,0), $K^\star = 0$, $K' = -10$



(d) Prey at (0,0), $K^\star = -10$, $K' = -10$



(e) Prey at (0,10), $K^\star = 10$, $K' = -10$



(f) Prey at (0,10), $K^\star = 7$, $K' = -8$

Figure 50: *Left*: prey movement is modeled as a diffusion process. *Right*: prey movement is modeled as a reaction-diffusion process. The free lunch curve (solid line) and the no-free lunch curve (dashed line) is shown for different positions of the center of prey density ('x').

start the sweep as a curve that intersects the prey center. If the fronts begin the sweep by hugging the prey, they scare away a lot of the prey and there are fewer left to capture during the rest of the sweep. As a result, the optimal lion-like fronts during a reaction-diffusion case are less aggressive and tend to hold back more than their diffusion case counterparts. In the case of Figure 50(b), the curve cannot "avoid" the prey any longer than its counterpart of the diffusion case, since the optimal curve in the diffusion case is already the curve that holds back the most, $Q(-10)$ (see Figure 50(a)).

In the case of the dolphin-inspired predator fronts, we notice that the optimal curve is always the curve that can place the predators in the "wing" positions (which capture the least prey) in the path of the initial ball of prey density. Figure 50(f) illustrates the only case when the optimal dolphin-like curve is not the curve $Q(-10)$; in fact, the optimal curve is $Q(-8)$ curve. This is makes sense because this is a reaction-diffusion case and by the time a curve reaches the prey center, the prey have extensively diffused and as a result, to eat more, the predators in the wing positions must spread out further. The result is a more straightened version of the diffusion counterpart obtained in Figure 50(e).

Due to the simplified models of prey response and constant length requirements placed on the predator fronts, we cannot expect exact replicas of natural systems. Even though our goal is bioinspiration, as mentioned before, we observe strong biomimicry for the case of the lion-inspired fronts. Dolphins in the wall method tend to charge in a straight line and this is not obtained in our simulations. We do however capture one interesting difference between lions and dolphins: the fact that lions rely on stalking their prey and dolphins do not. The maximum speed a lions can attain is $48 - 59$ Km/h, but they can only sustain this speed for a short distance [23]. The type of prey they hunt can usually outrun them and as a result, lions stalk their prey and charge only when they are within 30 m of their prey. On the other

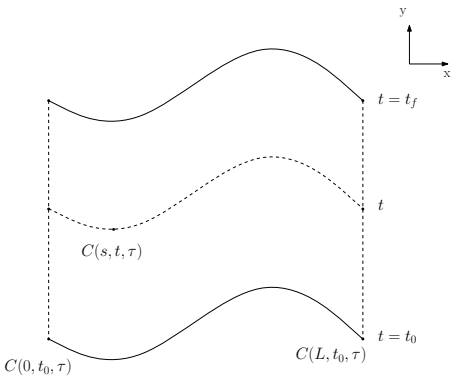


Figure 51: A curve front sweeps in the positive y direction with unit speed while maintaining its shape.

.

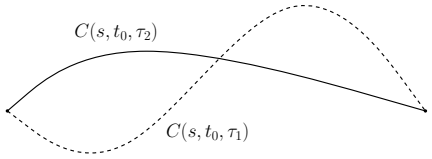


Figure 52: Curves evolving under the proposed algorithm share the same endpoints at $t = t_0$.

hand, dolphins tend to "drive" towards their for long distances (often until their drive is obstructed by the shore or boating activities [75]). This is captured by our results since the optimal lion-like fronts start in positions that are either the same distance or distance closer to the prey than the dolphin-like fronts.

7.4 Curve Evolution Model

In this section, we model the predator front as an arc length parameterized curve, define the energy over a curve, and produce an algorithm to increase the energy intake by deforming the shape of the curve. From the previous section, we retain the model used to describe the prey aggregation: a $2D$ time-varying density function denoted by $u(x, y, t)$, where $u : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}$ describes prey density at position (x, y) at time t.

7.4.1 Predator Fronts as Arc Length Parameterized Curves

We model the predator front as a curve of fixed shape sweeping through the aggregation of the prey, as shown in Figure 51. Without loss of generality, we assume that the front sweeps the area with unit speed and in the positive y direction while maintaining its shape. We define the energy intake of the front as the sum of the prey it sweeps over time and our goal is to find the best front shape that maximizes this energy intake.

Let the predator front be given by the curve, $C(s, t, \tau)$, where s is the arc length parameter, t denotes time, and $\tau \in \mathbb{R}$ parameterizes a family of time-varying curves. If we denote the total length of the curve by $L(\tau)$, then $s \in [0, L(\tau)]$. We will find the optimum shape of the front by evolving the curve C, using gradient ascent and moving in a direction that increases the energy intake. The main idea of the algorithm is to start with a curve shape $C(s, t, 0)$ and let it sweep the prey density from $t = t_0$ to $t = t_f$ and compute the energy intake, then deform the shape of the curve (with respect to τ) such that the energy intake is increased during the next sweep. We repeat these steps until the best curve shape is found.

It should be noted that with this curve-based model of the predator front, we are assuming a continuum of predators instead of the common agent-based model of foragers seen in [90]. Moreover, we assume that all curve shapes have identical endpoints, i.e. the endpoints of the curve stay the same regardless of the deformation in the shape of the curve (please see [38, 37] for details about this assumption).

We represent the energy-intake during a sweep of the curve, i.e. the amount of prey being "eaten" during a charge, as

$$\mathcal{E}(\tau) = \int_0^{t_f} \int_0^{L(\tau)} u(C(s, t, \tau), t)\ ds\ dt. \tag{43}$$

Our goal is to find the curve shape that maximizes this energy and we choose to use gradient ascent to update the curve shape, i.e. in such a way that the gradient of

$\mathcal{E}(\tau)$ with respect to τ is increased. The derivative of $\mathcal{E}(\tau)$ with respect to τ is given by

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \frac{d}{d\tau} \int_0^{L(\tau)} u(C(s,t,\tau),t)\ ds\ dt, \tag{44}$$

and to compute this derivative, we introduce a parameter $p \in [0,1]$ to replace the s parameterization of the curve with a parameterization that is not τ dependent. (For this substitution, we follow the method outlined in [53].) From the definition of arc length[1],

$$L(\tau) \ = \ \int_0^{L(\tau)} ds = \int_0^1 \|C_p(p,\tau)\|\ dp, \tag{45}$$

from which it follows that

$$ds = \|C_p(p,\tau)\|\ dp, \tag{46}$$

and

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \int_0^1 \left(u(C(s,t,\tau),t)\|C_p(p,\tau)\|\right)_\tau\ dp\ dt. \tag{47}$$

Using the chain rule, we get

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \int_0^1 (\ u_\tau\|C_p\| + u\|C_p\|_\tau\ )\ dp\ dt. \tag{48}$$

Notice that,

$$u_\tau = \nabla u \cdot C_\tau, \tag{49}$$

where $\nabla u(C(p,t,\tau),t)$ is the $2D$ spatial gradient of u. Also, we have that

$$\|C_p\|_\tau^2 = 2\|C_p\|\|C_p\|_\tau, \tag{50}$$

and

$$(C_p^T C_p)_\tau = 2C_{p\tau}^T C_p, \tag{51}$$

[1]For conciseness, we let f_x represent the partial derivative $\frac{\partial f}{\partial x}$ of a function $f(x,y)$ and denote the second-order partial derivative $\frac{\partial f}{\partial x \partial y}$ by f_{xy}.

where the superscript T denotes transpose. As a result,

$$\|C_p\|_\tau = C_{p\tau}^T \frac{C_p}{\|C_p\|} = C_{\tau p} \cdot \overrightarrow{T}, \tag{52}$$

where we note that the partial derivatives of C can be exchanged and we have introduced the unit tangent of the curve,

$$\overrightarrow{T}(p,\tau) = \frac{C_p}{\|C_p\|}. \tag{53}$$

Next, we substitute (49) and (52) into (48) and revert back to the s parameterization to obtain

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \int_0^{L(\tau)} \Big(\nabla u \cdot C_\tau + u C_{\tau s} \cdot \overrightarrow{T}(s,\tau) \Big) \; ds \; dt. \tag{54}$$

Using integration by parts,

$$\int_0^{L(\tau)} u C_{\tau s} \cdot \overrightarrow{T} \; ds = \Big[u C_\tau \cdot \overrightarrow{T} \Big]_{s=0}^{s=L(\tau)} - \int_0^{L(\tau)} C_\tau \cdot \Big(u \overrightarrow{T} \Big)_s \; ds. \tag{55}$$

Since we assume that the endpoints of the curve are fixed for all values of τ, $C_\tau(0,t,\tau) = 0$ and $C_\tau(L(\tau),t,\tau) = 0, \forall \tau, t \in [0,t_f]$. Equation (54) can now be written as

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \int_0^{L(\tau)} \Big(\nabla u \cdot C_\tau \; - \; C_\tau \cdot \Big(u \overrightarrow{T} \Big)_s \Big) \; ds \; dt. \tag{56}$$

Using the chain rule,

$$\begin{aligned} \Big(u \overrightarrow{T} \Big)_s &= u_s \overrightarrow{T} + u \overrightarrow{T}_s \\ &= \nabla u \cdot C_s + u \kappa \overrightarrow{N} \\ &= \nabla u \cdot \overrightarrow{T} + u \kappa \overrightarrow{N}, \end{aligned}$$

where we introduce two more intrinsic geometric properties of the curve: the unit normal $\overrightarrow{N}(s,\tau)$ and curvature $\kappa(s,\tau)$, through the relation $\overrightarrow{T}_s = \kappa \overrightarrow{N}$. Thus,

$$\begin{aligned} \frac{d\mathcal{E}(\tau)}{d\tau} &= \int_0^{t_f} \int_0^{L(\tau)} C_\tau \cdot \Big[- u\kappa \overrightarrow{N} + \nabla u - \big(\nabla u \cdot \overrightarrow{T} \big) \overrightarrow{T} \Big] \; ds \; dt \\ &= \int_0^{t_f} \int_0^{L(\tau)} C_\tau \cdot \Big[- u\kappa \overrightarrow{N} + \big(\nabla u \cdot \overrightarrow{N} \big) \overrightarrow{N} \Big] \; ds \; dt \\ &= \int_0^{t_f} \int_0^{L(\tau)} C_\tau \cdot \Big[\nabla u \cdot \overrightarrow{N} - u\kappa \Big] \overrightarrow{N} \; ds \; dt. \end{aligned} \tag{57}$$

Utilizing the fact that the only time-dependent functions are u and ∇u, we have

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{L(\tau)} C_\tau \cdot \left[\overrightarrow{N}^T \int_0^{t_f} \nabla u \; dt - \kappa \int_0^{t_f} u \; dt\right] \overrightarrow{N} \; ds. \tag{58}$$

With this expression of $d\mathcal{E}(\tau)/d\tau$, we will next present the curve flow algorithm used to characterize the most efficient predator front.

7.4.2 Curve Flow Algorithm

The main idea of the algorithm is to deform the shape of the curve so that the energy consumed by the predator front is increasing. To this end, our curve flow algorithm is inherently a gradient ascent algorithm and we propose the following curve evolution:

$$C_\tau := \left[\overrightarrow{N}^T \int_0^{t_f} \nabla u \; dt - \kappa \int_0^{t_f} u \; dt\right] \overrightarrow{N}, \tag{59}$$

with the result that

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{L(\tau)} \|C_\tau(s,\tau)\|^2 \; ds, \tag{60}$$

i.e., $d\mathcal{E}(\tau)/d\tau$ is non-negative, and the update rule for the curve becomes

$$C(s, 0, \tau_{next}) = C(s, 0, \tau) + (\tau_{next} - \tau) C_\tau(s, \tau), \tag{61}$$

except at the endpoints, where the curve shape does not change. Moreover, with this choice for curve evolution, we are not required to explicitly define τ; instead, it is driving the curve evolution by our choice of C_τ.

Note that the cost function given by (43) places no restriction on the length of the curve. The $\kappa \int_0^{t_f} u \; dt \overrightarrow{N}$ term, which represents a backward diffusion term in (59), can potentially generates infinitely long curves to increase the energy.

One way to address this would be to introduce a cost function that penalized the length of the curve. Consider the cost function, J, given by

$$J(\tau) = \mathcal{E}(\tau) - \rho L(\tau), \tag{62}$$

where $\mathcal{E}$ is the energy function given by (43) and ρ is some positive constant. We take the derivative with respect to τ and obtain,

$$\frac{dJ(\tau)}{d\tau} = \frac{d\mathcal{E}(\tau)}{d\tau} - \rho\frac{dL(\tau)}{d\tau}. \tag{63}$$

From (58), we have an expression for $d\mathcal{E}(\tau)/d\tau$ and what remains is to find an expression for $dL(\tau)/d\tau$. The total arc length of the curve can be written as

$$L(\tau) = \int_0^1 \|C_p\| \; dp, \tag{64}$$

and by taking the derivative with respect to τ, we have

$$\frac{dL(\tau)}{d\tau} = \int_0^1 C_{p\tau}^T \overrightarrow{T} \; dp \tag{65}$$

Noting the fact that the end points do not change with respect to τ and using integration by parts, we get

$$\frac{dL(\tau)}{d\tau} = -\int_0^{L(\tau)} C_\tau^T \overrightarrow{N} \kappa \; ds. \tag{66}$$

With (58) and (66), we can rewrite (63) as

$$\frac{dJ(\tau)}{d\tau} = \int_0^{L(\tau)} C_\tau \cdot \left[\overrightarrow{N}^T \int_0^{t_f} \nabla u \; dt - \kappa \left(\int_0^{t_f} u \; dt - \rho\right)\right] \overrightarrow{N} \; ds.$$

Consequently, for the evolution given by

$$C_\tau := \left[\overrightarrow{N}^T \int_0^{t_f} \nabla u \; dt - \kappa \left(\int_0^{t_f} u \; dt - \rho\right)\right] \overrightarrow{N}, \tag{67}$$

$dJ(\tau)/d\tau$ is non-negative. We will use the evolution given by (59) since for the examples used to illustrate the operation of the algorithm in the next section, the curve length remains bounded as the term $\overrightarrow{N}^T \int_0^{t_f} \nabla u \; dt \overrightarrow{N}$ dominates the term $\kappa \int_0^{t_f} u \; dt \overrightarrow{N}$.

For the curve flow algorithm, we are only required to specify the distribution of prey at each time; it is independent of movement laws used to describe the motion of the prey aggregation and the predator-prey interaction model. We apply the curve

flow algorithm to the two types of prey movement described in the Section 7.2: the purely diffusion based process and the reaction-diffusion based process.

The implementation of curve flow algorithm for a diffusion-based prey density is presented in Algorithm 1. By using (34), where the prey movement and predator positions are completely decoupled, the algorithm only requires us to calculate the prey density terms, $u(x, y, t)$ and $\nabla u(x, y, t)$, once. These values are stored and subsequently accessed each time the curve position is updated during a sweep. Also, notice that in Routine 2, the unit normal and the curvature is calculated only for the interior points on the curve. As a result, the two endpoints of the cure do not vary with τ, a requirement that was analytically useful in the derivation of $d\mathcal{E}(\tau)/d\tau$.

Note that for the reaction-diffusion case, for the curve $C(s, t, \tau)$, we now define the predator location, q, as follows:

$$q(x, y, t, \tau) = \begin{cases} 1 & \text{if } (x, y) \text{ on } C(s, t, \tau) \\ 0 & \text{otherwise} \end{cases}, \tag{68}$$

where $s \in \{0, \ldots, L(\tau)\}$. The algorithm for this case is similar to Algorithm 1, with the difference that density terms should be calculated inside Routine 3 at each time.

Algorithm 1 Curve flow algorithm

```
Specify initial prey distribution u(x, y, 0)
Calculate ∇u(x, y, 0)
for t = 0 : dt : t_f do
   Calculate u(x, y, t + dt)
   Calculate ∇u(x, y, t + dt)
end for
τ ⇐ 0
Generate N points to specify initial curve C(0, 0, 0)
L(0) ⇐ length of C(0, 0, 0)
Call Routine (2)
while ||C_τ(s, τ)|| > ε do
   τ ⇐ τ + 1
   L(τ) ⇐ length of C(0, 0, τ)
   Call Routine (2)
end while
```

Routine 2 Calculate C_τ

for $s = \frac{L(\tau)}{N}$ to $\frac{(N-1)}{N}L(\tau)$ **do**
  Calculate $\overrightarrow{N}(s,\tau)$
  Calculate $\kappa(s,\tau)$
  Call Routine (3)
  Calculate $C_\tau(s,\tau)$ according to (59)
  $C(s,0,\tau+1) \Leftarrow C(s,0,\tau) + \gamma C_\tau(s,\tau)$
  return $C(s,0,\tau+1)$
  return $C_\tau(s,0,\tau)$
end for

Routine 3 Calculate $\int_0^{t_f} u(C)$ and $\int_0^{t_f} \bigtriangledown u(C)$

$int\_u(C) \Leftarrow 0$
$int\_\bigtriangledown u(C) \Leftarrow 0$
for $t = 0 : dt : t_f$ **do**
  Calculate $u(C(s,t+dt,\tau))$
  Calculate $\bigtriangledown u(C(s,t+dt,\tau))$
end for
Calculate $int\_u(C)$
Calculate $int\_\bigtriangledown u(C)$
return $int\_u(C)$
return $int\_\bigtriangledown u(C)$

7.4.3 Simulations

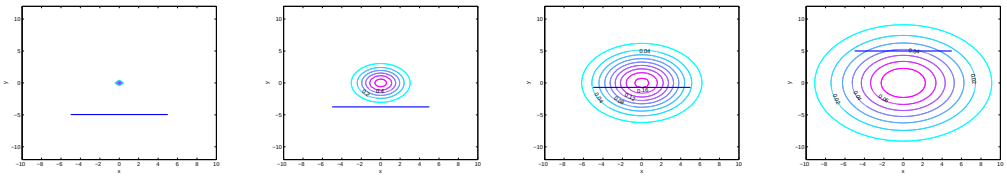
The foraging area is represented as a $2D$ mesh, where $x_{min} = -30$, $x_{max} = 30$, $y_{min} = -30$, $y_{max} = 30$, and the mesh spacing is $\Delta x = \Delta y = 0.5$. The initial prey density is a ball of radius 4 units with center located at $(0, 0)$. The diffusion process is numerically solved with thermal diffusivity set to 0.5. In fact, quantities like the unit normal, curvature, etc. are all calculated using finite difference methods.

For each τ, the resulting curve is swept through the prey density from $t_i = 0$ to $t_f = 20$, with a time step of $\Delta t = 0.005$. We use 21 data points to characterize a curve and the initial curve (at $\tau = 0$) is the straight line shown in Figure 53.

Figure 54 depicts the evolution of the curve front under our algorithm for both pure diffusion of prey (top) and reaction-diffusion case (bottom). The curves illustrated at Figures 54(c) and 54(f) respectively represent the optimal predator front for diffusion and diffusion-reaction cases.

The overall tendency of the curve (for both prey models) is to "hug" the center of the prey aggregation, which makes sense since during the duration of a sweep, the largest concentration of prey remains at the center. Since the end-points of the curve remain fixed under deformations, the most efficient curve resembles a bell-shaped curve (Figures 54(c) and 54(f)). Simulations also reveal that the shape of the most efficient predator front depends on the nature of its interaction with prey. For example, in the pure diffusion case of Figure 54(c), the agents in the "wing" positions drop back from their initial positions to sweep through more prey. On the other hand, in the reaction-diffusion case of Figure 54(f), these same agents cannot afford to drop back since prey will diffuse faster after sensing the dolphins in the center positions.

Figure 55 shows the effects of varying λ for prey movement given by (37). For "smarter" prey, i.e. prey that diffuse faster in the presence of predators, our resulting curve closely resembles a Bottlenose dolphin front, which is known to move "line abreast" during a charge through a school of fish [75].



(a) $t = 0$ (b) $t = 0.25t_f$ (c) $t = 0.5t_f$ (d) $t = t_f$

Figure 53: The curve at $\tau = 0$, a line, is swept through a prey density. The density, centered at $(0, 0)$ and diffusing according to (34), is represented with a contour map.

7.5 An Alternate Curve Evolution Model

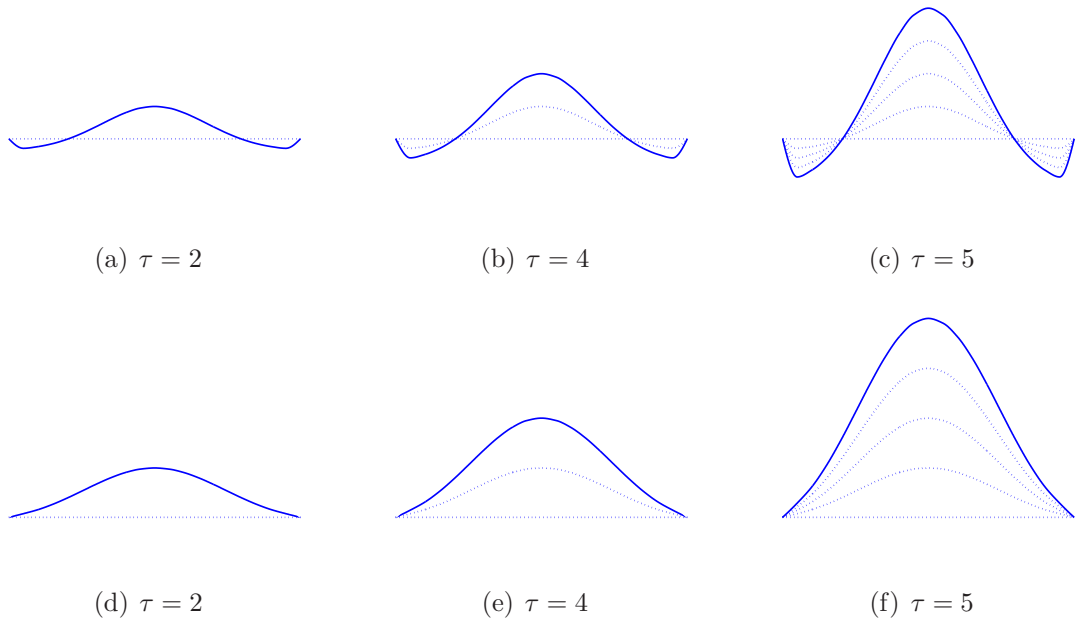
For the pure diffusion process, where the availability of prey at any given position does not depend on the position of the curve, the energy function given by (43), with the resulting curve evolution of (59), essentially produces a curve for the case of *instantaneous replacement* of prey. For example, consider the scenario where the agents configure themselves in a line, parallel to the direction of the sweep. According to (43), we would still maximize $u(C(s,t,\tau),t)$ along this curve during the sweep. We consider (43) to be at one end of the spectrum for defining the energy of the predator front. At the other end of this spectrum, during each sweep, we can formulate an energy function that represents the total flux of the vector $u(C(s,t,\tau),t)C_t(s)$ through the curve. Since the curve front sweeps in the positive y direction with unit speed, the flux lines of $u(C(s,t,\tau),t)C_t(s)$ points in the positive y direction. This new formulation for the energy-intake during a sweep of the curve is given by

$$\mathcal{E}(\tau) = \int_0^{t_f} \int_0^{L(\tau)} F \cdot \overrightarrow{N} \; ds \; dt, \tag{69}$$

where, $F = u(C(s,t,\tau),t)C_t(s)$.

7.5.1 Derivation

As outlined in Section 7.4, we take the derivative of the energy function with respect to τ, find an expression for C_τ, and then update the curve so that the gradient of $\mathcal{E}(\tau)$ with respect to τ is increased. Recall that to take this derivative, we use the p



(a) $\tau = 2$ (b) $\tau = 4$ (c) $\tau = 5$

(d) $\tau = 2$ (e) $\tau = 4$ (f) $\tau = 5$

Figure 54: Evolution of the predator front under the curve flow algorithm for prey diffusion (top) and reaction-diffusion (bottom).

parameterization of the curve, and after using the chain rule, we have

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \int_0^1 F_\tau \cdot \overrightarrow{N} \|C_p\| + F \cdot \overrightarrow{N}_\tau \|C_p\| + + F \cdot \overrightarrow{N} \|C_p\|_\tau \; dp \; dt. \tag{70}$$

Noting that $F_\tau = J_F C_\tau$, where J_F is the Jacobian of F, and using (52), we have

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \int_0^1 J_F C_\tau \cdot \overrightarrow{N} \|C_p\| + F \cdot \overrightarrow{N}_\tau \|C_p\| + + F \cdot \overrightarrow{N} C_{\tau p} \cdot \overrightarrow{T} \; dp \; dt. \tag{71}$$

From the definition of an outward normal, $\overrightarrow{N} = R\overrightarrow{T}$, where R is the $\frac{\pi}{2}$ clockwise rotation matrix,

$$\begin{aligned} \overrightarrow{N}_\tau &= (R\overrightarrow{T})_\tau \\ &= \frac{1}{\|C_p\|} R(C_{\tau p} - \overrightarrow{T} C_{\tau p} \cdot \overrightarrow{T}) \\ &= \frac{1}{\|C_p\|} R\overrightarrow{N} C_{\tau p} \cdot \overrightarrow{N} \\ &= \frac{-C_{\tau p} \cdot \overrightarrow{N}}{\|C_p\|} \overrightarrow{T}. \end{aligned} \tag{72}$$

(a) $\lambda = v_0/2$ (b) $\lambda = v_0$

Figure 55: Resulting curves for different values of λ.

Thus, we can now write (71) as

$$\frac{d\mathcal{E}(\tau)}{d\tau} = \int_0^{t_f} \int_0^{L(\tau)} J_F C_\tau \cdot \overrightarrow{N} - F \cdot (C_{\tau s} \cdot \overrightarrow{N}\,\overrightarrow{T}) + F \cdot \overrightarrow{N} C_{\tau s} \cdot \overrightarrow{T}\; ds\; dt, \tag{73}$$

when we switch back to the s parameterization. Furthermore, using integration by parts and by noting that $\overrightarrow{N}_s = R\overrightarrow{T}_s = \kappa \overrightarrow{T}$, $F_s = J_F C_s = J_F \overrightarrow{T}$, and $J_F C_\tau \cdot \overrightarrow{N} \in \mathbb{R}$, we have

$$\begin{aligned}
\frac{d\mathcal{E}(\tau)}{d\tau} &= \int_0^{t_f} \int_0^{L(\tau)} C_\tau \cdot J_F^T \overrightarrow{N} + (J_F \overrightarrow{T}) \cdot \overrightarrow{T} C_\tau \cdot \overrightarrow{N} - (J_F \overrightarrow{T}) \cdot \overrightarrow{N} C_\tau \cdot \overrightarrow{T}\; ds\; dt \\
&= \int_0^{t_f} \int_0^{L(\tau)} C_\tau \cdot \left[\overrightarrow{N}^T J_F \overrightarrow{N} + \overrightarrow{T}^T J_F \overrightarrow{T})\right] \overrightarrow{N}\; ds\; dt \\
&= \int_0^{t_f} \int_0^{L(\tau)} C_\tau \cdot \left[tr(J_F) \overrightarrow{N}\right]\; ds\; dt \\
&= \int_0^{t_f} \int_0^{L(\tau)} C_\tau \cdot (\nabla \cdot F \overrightarrow{N})\; ds\; dt,
\end{aligned} \tag{74}$$

where, $tr(\cdot)$ represents the trace of a matrix and $\nabla \cdot F$ denotes the divergence of the vector field F. With this expression for $d\mathcal{E}(\tau)/d\tau$, we choose the following curve evolution:

$$C_\tau = \int_0^{L(\tau)} \int_0^{t_f} \nabla \cdot F\; dt\; \overrightarrow{N}\; ds, \tag{75}$$

such that, $d\mathcal{E}(\tau)/d\tau$ is non-negative.

7.5.2 Simulations

Figure 56 shows the results of simulating a curve, initially given by a straight line, for the two the curve evolutions: one given by (59) and the other given (75). Three different initial density functions are considered for the prey and each density diffuses according to the pure diffusion process given by (34). The initial density is uniform in Figure 56(a); resembles a step pyramid in Figure 56(d); and essentially similar to Figure 56(a), but with a valley, in Figure 56(g).

For the prey density shown in Figure 56(g), notice that to maximize the energy, in the curve resulting from the original evolution model (Figure 56(h)), agents essentially line up behind one another to sweep through the valley longer. On the other hand, in this same location, the curve resulting from the evolution of (75) is U-shaped (Figure 56(i)).

7.6 Conclusions

In this chapter, inspired by the communal hunting techniques of Bottlenose dolphins and African lions, we first characterized the most efficient shape of the predator front for these two particular systems. The lion- and dolphin-inspired curves, i.e. the free lunch and the no free lunch curves were found by simulating a set of quadratic curves through the aggregation of prey, which was modeled using the population framework in biology. The free lunch curve maximized the total energy of the curve and the no free lunch curve maximized the energy of the agent feeding the least.

Next, we modeled the predator front as an arc length parameterized curve and using curve evolution techniques, we developed two curve deformation algorithms to maximize their corresponding cost functions. One cost function represented the total amount of prey swept by the curve and this resulted in a shape that allowed the instantaneous replacement of prey. The modified cost function considered the flux through the curve and the simulations were provided to illustrate the operation of

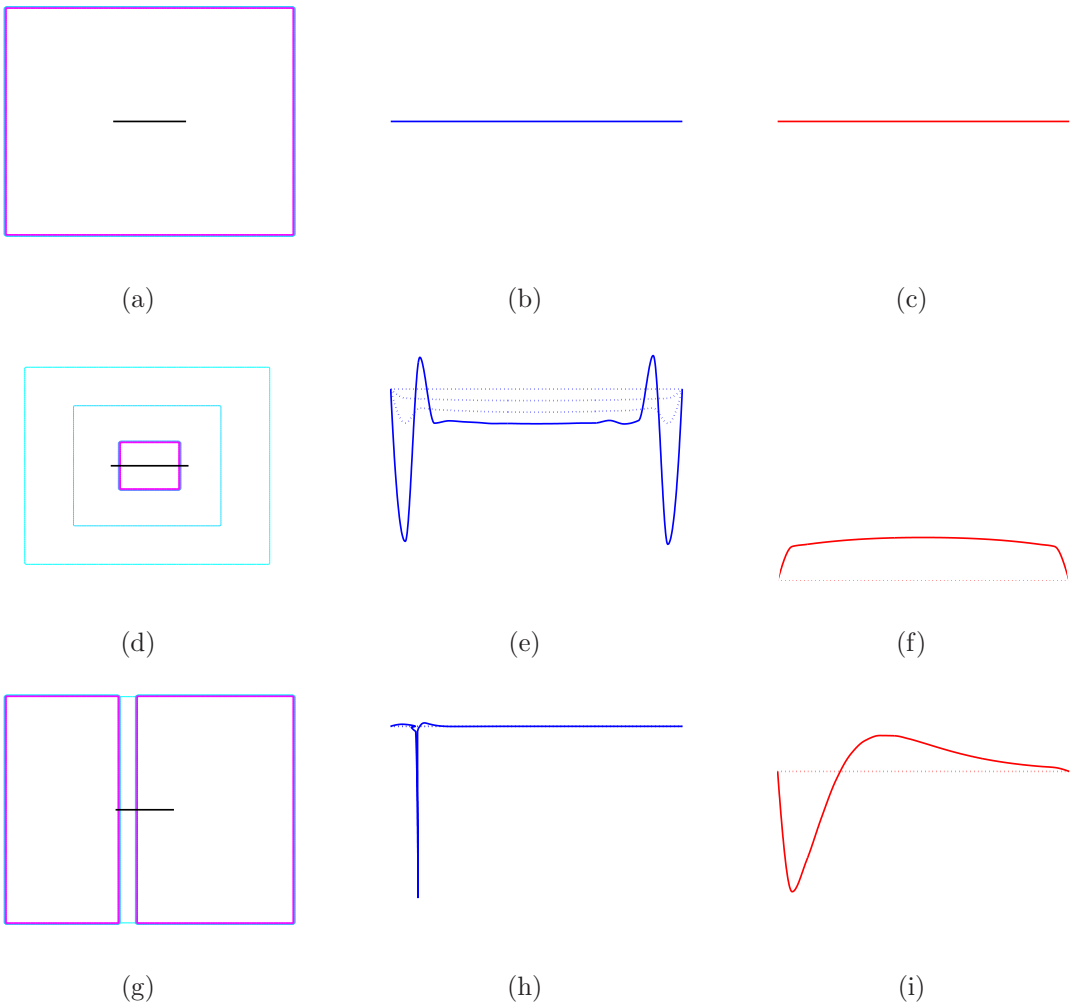


Figure 56: Each row has three figures: a contour map of the initial prey density accompanied by a straight line that depicts the curve at $\tau = 0$ on the left, the resulting curve using the evolution given by (59) in the middle, and the resulting curve using the evolution given by (75) on the right.

both of these algorithms. Two potential applications of this work: the design of a suction boom for surface oil skimming and the design of the multi-agent front for clearing littoral regions were discussed as well.

CHAPTER VIII

SUPPRESSION OF ENEMY AIR DEFENSES

In this chapter, we address the coordination aspects for a network of unmanned vehicles carrying out a search-and-destroy mission conceptualized by the US Navy known as the suppression of enemy air defenses (SEAD) mission. It requires a team of surveillance vehicles to coordinate with a team of combat vehicles to successfully search, identify, and neutralize threats. The targeted strategy is obtained by tailoring the multi-level alliance model of Bottlenose dolphins produced in Chapter 3.

8.1 Introduction

The description of SEAD in [1] involves a heterogeneous network with three types of agents: high-altitude and -endurance unmanned aerial vehicles (HAE UAVs), medium-altitude and -endurance unmanned aerial vehicles (MAE UAVs), and uninhabited combat air vehicles (UCAVs). The HAE UAVs and MAE UAVs provide surveillance; in fact, together they form the intelligence, surveillance, and reconnoissance (ISR) team. The idea is to use a higher-altitude detector vehicle to "cue" a lower-altitude examiner vehicle for accurate identification of threats. The HAE UAV is usually equipped with a low-resolution imaging capability to detect areas of interest. It alerts the slower moving MAE UAV, which is equipped with a high-resolution imaging capability to search these areas and identify potential threats. Thus, the two vehicles cooperate to achieve "final target confirmation" and information about the threat (its nature, location, etc.) is passed on to a team of UCAVs that are responsible for neutralizing the threat. Teams of UCAVs can confuse the radar systems associated with threats [7] and as such, they provide more effective combat support to ISR teams than a single UCAV.

Arslan et al. [4] and *Beard et al.* [7] are examples of some previous works in the general area of target elimination using multiple vehicles. In [4], a game-theoretic negotiation policy is developed that optimally assigns vehicles to targets. Moreover, a utility function is designed so that locally interacting vehicles meet a global performances criterion. In [7], UAVs are assigned to known target locations and the subproblems addressed in the work include target assignment, path planning, and trajectory following. There has been significant interest in the SEAD mission in particular (for a representative sample, see [20, 24, 77]). However, these research efforts focus on the combat aspects of the SEAD mission and as a result, the network in these works only consist of UCAVs. In [20], linear programming is used for task allocation of UCAVs. Path finding using potential fields for UCAVs is presented in [24], and in [77], a simulation framework is introduced to test control algorithms for cooperative UCAVs. In this chapter, we consider the coordination of both the ISR team members and members of the UCAV teams. Also, threats are neutralized only through the cooperation of ISR and UCAV teams; as a result, we develop a two-level coalition formation strategy.

A framework for producing multi-level alliances was addressed in Chapter 3, where using tools from hybrid systems and embedded graph grammars (EGGs), we produced a model that captured the alliance forming behavior of male Bottlenose dolphins. In our model, male dolphin agents produced clusters of pair and triplets (first-order alliance) and two clusters combined to form a higher-order team (second-order alliance) for a specific task, i.e. to steal/defend female dolphin agents. In this chapter, we tailor the model developed in Chapter 3 for the SEAD mission.

The remainder of this chapter is organized as follows: in Section 8.2 we introduce notations and the exploration strategy used by the ISR team members. Based on the model of dolphin alliances, higher-order teams are built in provided in Section 8.3. Simulations of the bio-inspired coordination strategy are shown in Section 8.4 and

conclusions are provided in Section 8.5.

8.2 Setup

8.2.1 Notations

EGGs were previously introduced in Chapter 3 to model the multi-level alliance forming behavior of Bottlenose dolphins. Recall that, for the vertex-labeled graph $\mathcal{G} = (\mathcal{V},\ \mathcal{E},\ l,\ \Sigma)$, $\mathcal{V}$ is the set of agents (i.e. vehicles), $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges, and the function l assigns a label from the label set, Σ, to the agents in the network. Furthermore, each rule in the set of transition rules, Φ, is given by a pair $r = (L \rightharpoonup R)$, where L and R are subgraphs. Also, if the state-dependent guard condition associated with rule r is `true`, then the application of the rule r on the original subgraph L produces the subgraph, R.

The following are some set notations:

- $\mathcal{H} = \{h_1, \ldots, h_{N_H}\}$ is the set of all HAE UAVs,
- $\mathcal{M} = \{m_1, \ldots, m_{N_M}\}$ is the set of all MAE UAVs,
- $\mathcal{C} = \{c_1, \ldots, c_{N_C}\}$ is the set of all UCAVs,
- $\mathcal{I}$ is the set of all ISR teams,
- $\mathcal{J}_1$ is the set of all UCAV pairs,
- $\mathcal{J}_2$ is the set of all UCAV triplets,
- $\mathcal{S}_1$ is the set of all higher-order teams of size 4,
- $\mathcal{S}_2$ is the set of all higher-order teams of size 5,
- $\mathcal{N}_i$ denotes the neighborhood set of vehicle i, and
- $\mathcal{T} = \{t_1, \ldots, t_{N_T}\}$ is the set of all targets.

Thus, the set of all vehicles is given by $\mathcal{V} = \mathcal{H} \cup \mathcal{M} \cup \mathcal{C}$. The edge set, $\mathcal{E}$ is defined through the nearest-neighbor rule of Chapter 3 and consequently, $j \in \mathcal{N}_i$ if $(i,\ j) \in \mathcal{E}$, where $i,\ j \in \mathcal{V}$. Also, each HAE UAV, MAE UAV, and UCAV is initially assigned the label h^0, m^0, and c^0, respectively, and these superscripts denote the fact that initially, the vehicles are not part of a coalition.

8.2.2 Exploration

Let $\mathcal{A}$ be a set of coordinates and assume that the exploration task is complete once the HAE UAVs have visited these coordinates. The idea is that by visiting all the coordinates in $\mathcal{A}$, through the detection range of their sensors, the HAE UAVs cover the entire search area. Thus, the information of a sub-area - the size corresponding to the sensing range of HAE UAVs - is embedded in the information gathered from the coordinates associated with that sub-area and the together, these sub-areas span the entire search area. When the HAE UAV detects a threat in one of these coordinates, it passes this information to a MAE UAV; more specifically, its ISR team member.

This MAE UAV is now responsible for combing the sub-area and we assume that there is a set of coordinates associated with each coordinate in $\mathcal{A}$ and by traversing these coordinates, the sub-area is completely explored. We denote $\mathcal{A}'_i$ to be the set of coordinates that must be explored by the MAE UAV in charge of searching the node $i \in \mathcal{A}$ for target confirmation. Furthermore, $\mathcal{A}'_i \cap \mathcal{A}'_j = \varnothing,\ \forall\ i, j \in \mathcal{A},\ i \neq j$.

Each HAE UAV uses the functions $\texttt{h.explored} : \mathcal{H} \times \mathcal{A} \to \{0, 1\}$ and $\texttt{h.detect} : \mathcal{H} \times \mathcal{A} \to \{0, 1\}$ to classify the coordinates in $\mathcal{A}$. HAE UAV i will visit an unexplored coordinate $j \in \mathcal{A}$, i.e., $\texttt{h.explored}(i, j) = 0$ and if it senses a threat in the area associated with that coordinate, it will set $\texttt{h.detect}(i, j) = 1$. Similarly, the MAE UAVs use the function $\texttt{m.explored} : \mathcal{M} \times \mathcal{A}'_i \to \{0, 1\}$, where $i \in \mathcal{A}$.

In the next section, we will first specify the rules to form surveillance and combat teams, and subsequently, present rules to form higher-order teams that consist of a

combination of the two.

8.3 Heterogeneous Unmanned Vehicle Teams

8.3.1 ISR Teams

The ISR team is initiated once an HAE UAV detects a threat while traversing the coordinates in $\mathcal{A}$. It requests the nearest MAE UAV and if this MAE UAV accepts the request, the two vehicles form an ISR team. The formation of an ISR team is represented by the following transition rule:

$$\Phi^1 = \ h^0 \quad m^0 \ \rightharpoonup \ h^1 - m^1 \ \ (r_1), \tag{76}$$

where we define a label set $\Sigma^1 = \{h^0, m^0, h^1, m^1\}$. Moreover, for two vehicles i and j, where $l(i) = h^0$ and $l(j) = m^0$, if the guard condition for r_1 is `true`, then the application of rule r_1 relabels the two vehicles and the outcomes are $l(i) = h^1$, $l(j) = m^1$, and $(i, j) \in \mathcal{I}$. The guard condition associated with r_1 is `true` if 1) vehicle i sends a request to vehicle j and 2) vehicle j accepts this request.

Vehicle i will send a request to vehicle j if vehicle j is the nearest vehicles to i labeled m^0 and $\texttt{h.detect}(i, k) = 1$, where $k \in \mathcal{A}$. We also design a method by which an MAE UAV makes a selection when presented with multiple HAE UAV requests. Let the function $\texttt{h.quality}(i, k) : \mathcal{H} \times \mathcal{A} \rightarrow [0, 1]$ denote the quality of the detection made by HAE UAV i regarding coordinate k. Consider the HAE UAV m that simultaneously sends a request to vehicle j, but regarding coordinate n. If $\texttt{h.quality}(i, k) > \texttt{h.quality}(m, n)$, it implies that there is less uncertainty regarding the possibility of a threat being located in the sub-area associated with coordinate k. In this case, according to our model, vehicle j will accept the request sent by the HAE UAV i. The guard condition for r_1 evaluates to `true` and subsequently, vehicle j will begin exploring the coordinates in $\mathcal{A}'_k$. Furthermore, we assume that the HAE UAV initiates a holding pattern over the coordinate k until it is relabeled to h^0 (this transition rule is described later).

8.3.2 Combat Teams

There is no specific size for the UCAV teams mentioned in [1]. In [7], multiple combat-capable vehicles are required to arrive at the target location at the same time to confuse its defense system, but it is conceivable that too many vehicles can deteriorate the group's ability to coordinate effectively (this topic was previously addressed in Chapter 5). Based on the model of first-order alliances of dolphins presented in Chapter 3, we restrict the size of UCAV teams to either pairs or triplets. The rules for building UCAV teams are as follows:

$$\Phi^2 = \begin{cases} c^0 \quad c^0 \rightharpoonup c^1 \text{——} c^1 \quad (r_2), \\ \quad c^1 \qquad\quad\; c^2 \\ \quad\diagup \qquad \rightharpoonup \quad \diagup\ \diagdown \\ c^1 \quad c^0 \quad c^2 \text{——} c^2 \quad (r_3). \end{cases} \tag{77}$$

The label set used for UCAV teams is $\Sigma^2 = \{c^0, c^1, c^2\}$. We assign capabilities to individual UCAVs and this measure of capability can be thought of as a function of the state of the vehicle. For example, it can be designed as a function of either the vehicle's fuel consumption, onboard weapon systems, or health. Let the capability of combat vehicle i be given by the non-increasing function $\texttt{c.cap}(i) \in \mathbb{R}$, and in our model, an UCAV will be available to form a combat team if $\texttt{c.cap}(i) \geq \xi_c$, where ξ_c is a positive constant. The idea behind this formulation is that a UCAV must be "fit" in order to join a combat team. Groups are formed when the sum of the individual capabilities of the members increases a certain threshold, denoted by a positive constant ξ_t, and the capabilities of each member is greater than ξ_c.

When the guard condition of r_2 is `true`, for two vehicles i and j labeled c^0, the outcome of applying r_2 is the formation of a combat pair - $(i, j) \in \mathcal{J}_1$, $l(i) = c^1$, and $l(j) = c^1$. The guard condition of r_2 is true if $\texttt{c.cap}(i) \geq \xi_c$, $\texttt{c.cap}(j) \geq \xi_c$, and $\texttt{c.cap}(i) + \texttt{c.cap}(j) \geq \xi_t$. A triplet is formed through the addition of an agent to an existing pair (an idea previously introduced in Chapter 3). Thus, for three agents, i, j, and k, where $(i, j) \in \mathcal{J}_1$, $l(i) = c^1$, $l(j) = c^1$, and $l(k) = c^0$, then the application

of r_3 forms the triplet (i, j, k), where $(i, j, k) \in \mathcal{J}_2$, $l(i) = c^2$, $l(j) = c^2$, and $l(k) = c^2$. The guard condition of r_3 is true if $\texttt{c.cap}(i) \geq \xi_c$, $\texttt{c.cap}(j) \geq \xi_c$, $\texttt{c.cap}(k) \geq \xi_c$, and $\texttt{c.cap}(i) + \texttt{c.cap}(j) + \texttt{c.cap}(k) \geq \xi_t$.

In the simulations shown in Section 8.4, the capability corresponds to onboard weapon systems and after a team neutralizes a threat, the capability of the members in the group decreases. Also, in our setup, the threshold ξ_t is chosen in such a way that any UCAV team is capable of neutralizing at least one threat.

The breakdown of a combat team can be described by the following rules:

$$\Phi^3 = \begin{cases} c^1 \text{---} c^1 \rightharpoonup c^0 \quad c^0 & (r_4), \\ \begin{array}{c} c^2 \\ / \quad \backslash \\ c^2 \text{---} c^2 \end{array} \rightharpoonup \begin{array}{c} c^0 \\ \\ c^0 \end{array} & (r_5). \end{cases} \tag{78}$$

The rules r_4 and r_5 have the same guard condition, which evaluates to `true` when either the capability of at least one vehicle in the combat team drops below ξ_c or the sum of the capabilities drops below ξ_t.

8.3.3 Higher-Order Teams

Consider an ISR pair searching the coordinate $k \in \mathcal{A}$. While the MAE UAV visits all the coordinates in $\mathcal{A}'_k$, the HAE UAV maintains a holding patter around coordinate k. After the MAE UAV completes its exploration, it shares the number of targets identified with its ISR member. We denote the number of targets by $\texttt{m.num}(i, j)$, where $\texttt{m.num} : \mathcal{M} \times \mathcal{A} \to \mathbb{N}_0$ describes the number of targets identified by MAE UAV i after exploring all the coordinates in $\mathcal{A}'_j$. If at least one target is detected, both members of the ISR team initiate a request to form a higher-order team.

We introduce an artificial label set $\sigma^1 = \{I, J_1, J_2\}$, where I, J_1, and J_2 represent the ISR team, the combat pair, and the combat triplet, respectively (this label set is not part of the vertex-labeled graph). In Chapter 3, this use of a shorthand is introduced to replace a coalition with a single label, known as a "supernode," and

details are presented to specify the possible interactions between supernodes. The rules for building higher-order teams is presented below.

$$\Phi^4 = \begin{cases} I \quad J_1 \rightharpoonup I^4 \text{---} J_1^4 \quad (r_6), \\ I \quad J_2 \rightharpoonup I^5 \text{---} J_2^5 \quad (r_7), \end{cases} \tag{79}$$

where we introduce labels from another artificial label set $\sigma^2 = \{I^4, I^5, J_1^4, J_1^5, J_2^4, J_2^5\}$ and the superscripts refer to the size of the resulting higher-order team. Consider the coalition labeled I, where the two vehicles in this coalition are actually labeled h^1 and m^1. When this coalition joins the combat team labeled J_1, members of the ISR teams gain two more coalition members; as such, they are relabeled as h^3 and m^3, and this ISR team is represented in the rule set as I^4. We introduce one more label set: $\Sigma^3 = \{h^3, m^3, c^3, h^4, m^4, c^4\}$; therefore, the label set Σ of the entire network is given by $\Sigma = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3$.

The guard condition for r_6 evaluates to `true` if the combat team (m, n) accepts the request sent by the ISR team (i, j). The ISR team sends a request to the nearest combat team if $\mathtt{m.num}(j, k) > 0$. Given a choice between two combat teams, according to this model, the ISR team prefers the larger team. The combat team accepts this request if it is available, i.e. $(m, n) \notin \mathcal{S}_1$. Thus, for four vehicles, i, j, m, and n, where $l(i) = h^1$, $l(j) = m^1$, $l(m) = c^1$, $l(n) = c^1$, and $(i, j) \in \mathcal{I}$, $(m, n) \in J_1$, the application of r_4 yields $(i, j, m, n) \in \mathcal{S}_1$ and the vehicles are relabeled as follows: $l(i) = h^3$, $l(j) = m^3$, $l(m) = c^3$, and $l(n) = c^3$.

The guard condition for r_7 evaluates to `true` if the combat team (m, n, o) accepts the request sent by the ISR team (i, j). For the vehicles, i, j, m, n, and o, where $l(i) = h^1$, $l(j) = m^1$, $l(m) = c^2$, $l(n) = c^2$, $l(o) = c^2$, and $(i, j) \in \mathcal{I}$, $(m, n, o) \in J_2$, the application of r_5 yields $(i, j, m, n, o) \in \mathcal{S}_2$ and the vehicles are relabeled as follows: $l(i) = h^4$, $l(j) = m^4$, $l(m) = c^4$, $l(n) = c^4$, and $l(o) = c^4$.

8.3.4 Breakdown

The MAE UAV in the higher-order team shares both the number of targets and their locations with the combat team. In turn, the combat team shares the status of these targets with the ISR team. The higher-order team is disbanded based on two types of conditions - one associated with the state of the threats and the other is associated with the state of the combat team. The rules are described as follows:

$$\Phi^5 = \begin{cases} I^4 - J_1^4 \rightharpoonup h^0 \; m^0 \; J_1 \quad (r_8), \\ I^4 - J_1^4 \rightharpoonup I \; c^0 \quad c^0 \quad (r_9), \\ I^5 - J_2^5 \rightharpoonup h^0 \; m^0 \; J_2 \quad (r_{10}) \\ I^5 - J_2^5 \rightharpoonup I \; c^0 \; c^0 \; c^0 \; (r_{11}). \end{cases} \tag{80}$$

Thus, the rule set Φ of the entire network is given by $\Phi = \Phi^1 \cup \Phi^2 \cup \Phi^3 \cup \Phi^4 \cup \Phi^5$. The guard conditions associated with rules r_8 and r_{10} are `true` if the combat team clears $\texttt{m.num(i,j)}$ threats (MAE UAV i is part of the higher-order team and $j \in \mathcal{A}$). In both these case, the ISR team is broken and HAE UAV returns to exploring the coordinates in $\mathcal{A}$.

The guard conditions associated with rules r_9 and r_{11} are `true` if the combat team dissolves before all threats are cleared. In these two cases, the ISR team remains intact and reinitiates the request to form a higher-order team.

Notice that the breakdown of ISR teams depends on the disbanding of the corresponding higher-order team. This is due to the fact that the formation rules of ISR teams depend on the existence of threats and once this threat is cleared, the higher-order team is broken, along with the ISR team.

8.4 Simulations

The operation of our bio-inspired strategy is shown in Figure 57 and continues in Figure 58. The search area is partitioned into 16 cells and a dot in the center of a cell represents a coordinate in $\mathcal{A}$. A cell is highlighted when explored by at least one

HAE UAV. There are four threats (dot with a ring) in the search area; two of which are located in the same cell.

The simulation includes two HAE UAVs (large, planes), three MAE UAVs (small, planes), and five UCAVs (wedges). An exploring HAE UAV is shown with a sensing footprint around it. When an HAE UAV detects a threat (no sensing footprint) it sends a request to form an ISR team. The MAE UAV that accepts this request (shown with a sensing footprint around it) begins searching the cell and the HAE UAV begins a holding pattern over that cell. Once the MAE UAV searches the entire cell and confirms a threat, the ISR team sends a request to form a higher-order team. Members of combat teams are indicated by bold lines between them. A combat team that accepts this request locks on to the location of the threat(s) specified by the MAE UAV (shown with a dashed line between the team and the threat).

Notice that according to our model, the ISR team prefers to form a higher-order team with the larger UCAV team (Figure 57(e)). Moreover, since the capability of combat vehicles decrease after neutralizing threats, notice that in Figure 58(d), after neutralizing all the threats, the combat teams have all broken into single vehicles.

The mission is complete when there all the coordinated in $\mathcal{A}$ have been explored by the HAE UAVs. At this point, the vehicles return to their initial positions (these locations are not shown) (Figure 58(e)).

8.5 Conclusions

We tailored the model of dolphin alliances to develop a coordination strategy for a network of heterogeneous unmanned vehicles to form multi-level coalitions. We provided a broad framework for describing the coordination rules in a way that allows specific tactics to be "plugged in." The framework allows for the guard conditions to be easily replaced by a more sophisticated set of conditions, e.g., utilizing game theoretic concepts like regret matching [4] when assigning targets.

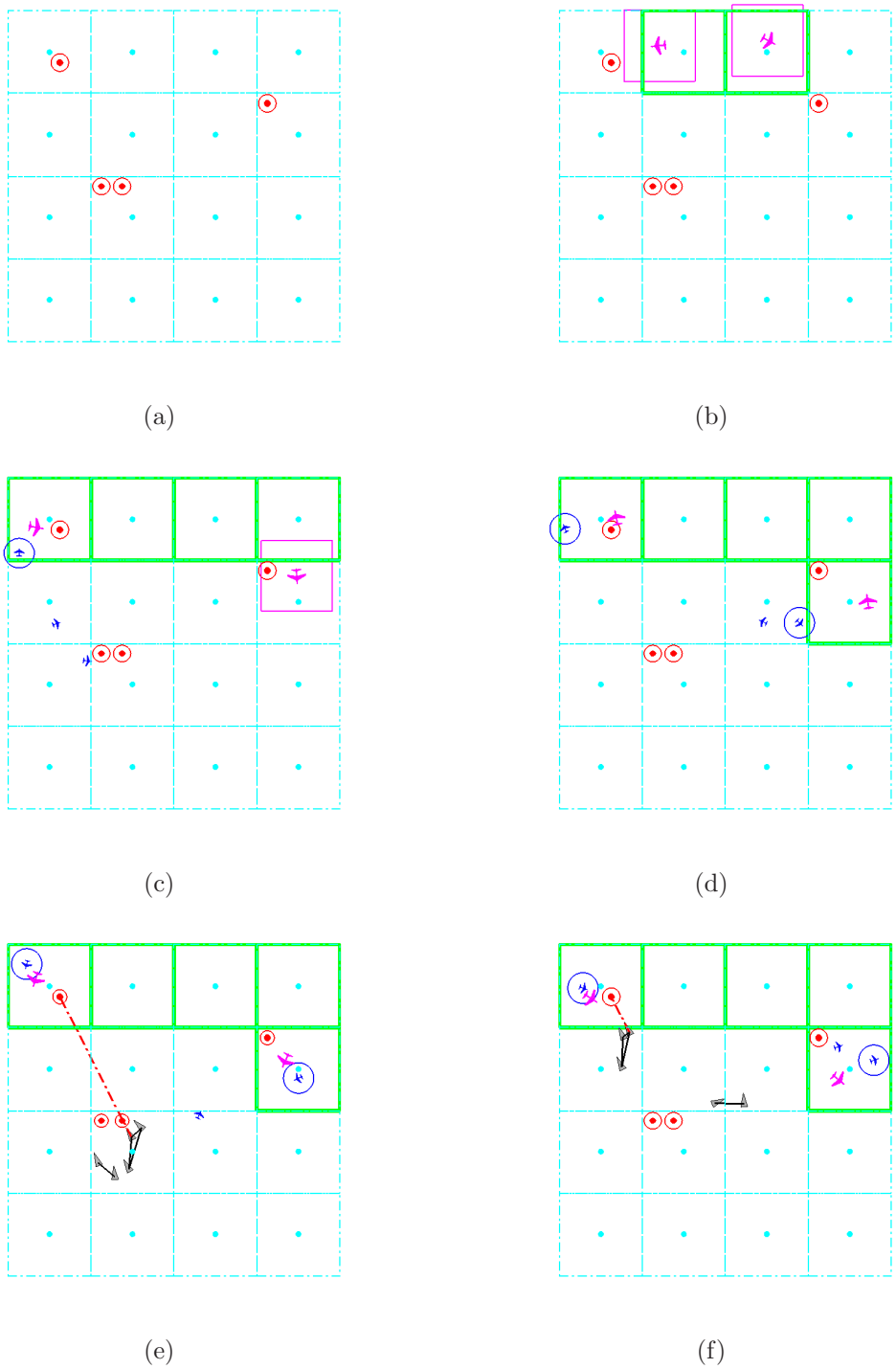
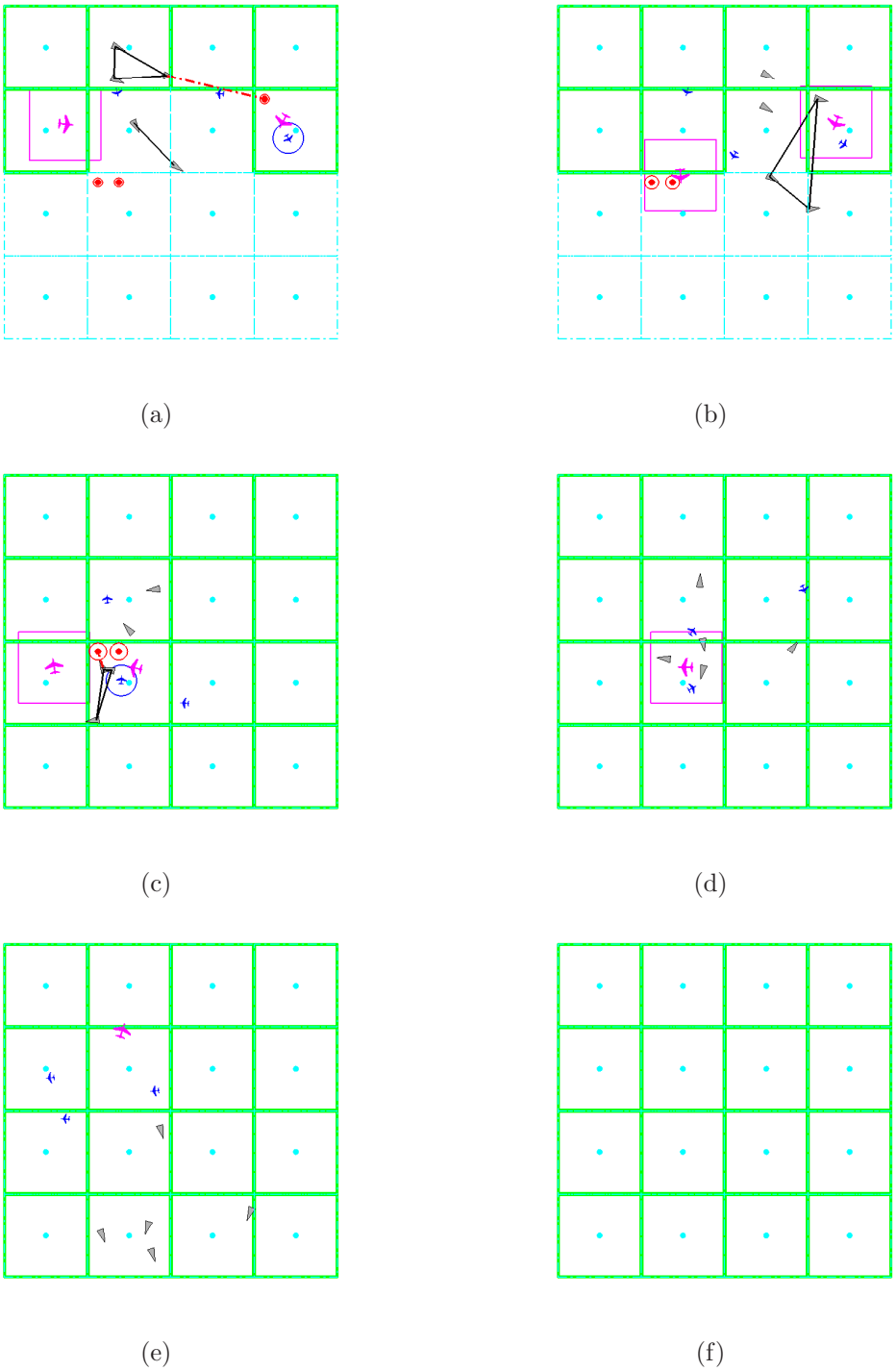


Figure 57: The bioinspired strategy to coordinate a heterogeneous team of vehicles during the SEAD mission is shown.



(a) (b) (c) (d) (e) (f)

Figure 58: Simulation of Figure 57 continues here.

CHAPTER IX

CONCLUSIONS

9.1 Conclusions

We developed biologically inspired coordination strategies for multi-agent systems. We addressed the following areas in our approach to bioinspiration: models, algorithms, and applications. The contributions from each area are summarized below.

1. **Models.** In the first part of the thesis, we produced agent-based models of social behaviors observed in biological systems. We focused on small-scale biological systems; particularly, Bottlenose dolphins and African lions. The goal of these models were twofold: capture the underlying biological phenomenon, yet remain simple, and as such, remain amenable to analysis. The work in this part of the thesis included a model of the alliance forming behavior of dolphins, a foraging model of dolphins, and a model of the social structures of lions. Simulation were provided to illustrate the richness of these models.

2. **Algorithms.** In the second part, we developed tunable algorithms based on the simple biological models produced in the first part of the thesis. Moreover, these algorithms were developed in the context of specific multi-agent coordination tasks. First, we considered the containment problem; more specifically, the containment of a non-cooperative group of mobile agents. An algorithm was developed - based on the horizontal carousel method used by dolphins to catch fish - for the perpetual confinement of fish agents using a single dolphin agent. A potential application for this work was also identified - the containment of service robots using a single mobile agent. In another work, inspired by the wall method, where dolphins form a predator front to charge through fish, we

developed an algorithm to characterize the most efficient shape of a foraging front. The foraging front formed by the agents was modeled as a curve and curve evolution techniques were used to optimize the shape of this front. Application domains were also discussed and these included autonomous mine-clearing and oil-spill recovery using a flexible suction boom and a group of robots.

3. **Applications.** In the final part of the thesis, we demonstrated how simple biological models can be used for specific problems in engineered systems. For a target application, the US Navy suppression of enemy air defenses mission, we tailored the alliance forming model of dolphins to produce coordination strategies for a network of heterogenous unmanned vehicles. Simulations of the coordination strategy were shown where teams of surveillance vehicles cooperated with teams of combat vehicles to successfully clear threats from an area.

9.2 Future Directions

The work in this thesis serves as a foundation for the investigation of several other research topics. One extension to this work includes producing a library of tunable algorithms which can be deployed in numerous target applications. One specific example was provided in Chapter 8, the suppression of enemy air defenses, and other potential applications include boundary protection using a network of robots and coalition-based resource allocation in mobile sensor networks.

In this thesis, the agent-based biological models were designed to contribute to the coordination challenges in engineered systems. An interesting line of inquiry would be to apply these models to contribute to the field of biology. Consider the main result of Chapter 3, which states that in a network that consists of three types of first-order alliances, an alliance cannot steal back a lost female dolphin without anything else changing in the network. There is evidence of this behavior in actual dolphin herds, which suggests that our alliance model - although simple - is quite

expressive. Instead of using the biological models for engineered applications, it is worth exploring the effectiveness of these model when incorporated with biological field data. Some challenges include modifying the models based on observed social behaviors and presenting analytical results in the context of biological systems. In essence, we would reverse the direction of the inspiration used in this thesis and use tools from multi-agent systems domain to offer insights into the natural systems domain.

REFERENCES

[1] *Autonomous Vehicles in Support of Naval Operations*. Washington, D.C.: The National Academic Press, 2005.

[2] "Biologically-inspired approaches for team and coalition adaptation of heterogeneous unmanned systems for surveillance over large and complex areas," in *Office of Naval Research Broad Agency Announcement 07-036*, 2008.

[3] Andrews, B., Passino, K., and Waite, T. A., "Social foraging theory for robust multiagent system design," *IEEE Trans. on ASE*, vol. 4, pp. 79–86, 2007.

[4] Arslan, G., Marden, J., and Shamma, J., "Autonomous vehicle-target assignment: A game theoretical formulation," *ASME Journal of Dynamic Systems, Measurement, and Control*, pp. 584–596, Sep. 2007.

[5] Balch, T., "The impact of diversity on performance in multi-robot foraging," in *Proc. Third Conf. on Autonomous Agents*, (Seattle, WA), pp. 92–99, 1999.

[6] Balch, T., "The impact of diversity on performance in multirobot foraging," in *Autonomous Agents*, 1999.

[7] Beard, R., McLain, T., Goodrich, M., and Anderson, E., "Coordinated target assignment and intercept for unmanned air vehicles," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 911–922, Dec. 2002.

[8] Berman, S., Halász, A., Kumar, V., and Pratt, S., "Algorithms for the analysis and synthesis of a bio-inspired swarm robotic system," in *Swarm robotics (Lecture Notes in Computer Science 4433)* (Sahin, E., Spears, W. M., and Winfield, A. F. T., eds.), pp. 56–70, Springer-Verlag, 2007.

[9] Bygott, J., Bertram, B., and Hanby, J., "Male lions in large coalitions gain reproductive advantages," *Nature*, vol. 282, pp. 839–841, 1979.

[10] Cao, Y. U., Fukunaga, A., and Kahng, A., "Cooperative mobile robotics: Antecedents and directions," *Autonomous Robots*, vol. 4, no. 1, pp. 7–27, 1997.

[11] Chan, T. and Vese, L., "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[12] Conner, R., Heithaus, M., and Barre, L., "Complex social structure, alliance stability and mating access in a bottlenose dolphin 'super-alliance'," *Proceedings of the Royal Society of London Series B*, vol. 268, pp. 263–267, Feb. 2001.

[13] Conner, R., Smolker, R., and Richards, A., "Two levels of alliance formation among male bottlenose dolphins (tursiops sp.)," in *Proc. Natl. Acad. Sci.*, pp. 987–990, Feb. 1992.

[14] Connor, R., Heithaus, M., and Barre, M., "Superalliance of bottlenose dolphins," *Nature*, vol. 371, pp. 571–572, 1999.

[15] Cortes, J., Martinez, S., and Bullo, F., "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Transactions on Automatic Control*, vol. 51, no. 8, pp. 1289–1298, 2006.

[16] Cousteau, J. and Diolé, P., *Dolphins*. Garden City, NY: Doubleday and & Company, Inc, 1975.

[17] Couzin, I., "Behavioral ecology: social organization in fission-fusion societies," *Current Biology*, vol. 16, pp. 169–171, 2006.

[18] Couzin, I. and Franks, N., "Self-organized lane formation and optimized traffic flow in army ants," *Proceedings of the Royal Society of London Series B*, vol. 270, pp. 139–146, 2003.

[19] Couzin, I., Krause, J., Franks, N., and Levin, S., "Effective leadership and decision making in animal groups on the move," *Nature*, vol. 433, pp. 513–516, 2005.

[20] Darrah, M., Niland, W., and Stolarik, B., "Multiple uav dynamic task allocation using mixed integer linear programming in a sead mission," in *Infotech@Aerospace*, (Arlington, VA), Sep. 2005.

[21] Egerstedt, M., "Graph-theoretic methods for multi-agent coordination," in *ROBOMAT*, 2007.

[22] Egerstedt, M. and Hu, X., "Formation constrained multi-agent control," *IEEE Transactions on Robotics and Automation*, vol. 17, pp. 947–951, Dec. 2001.

[23] Estes, R., *Behavior Guide to African Mammals*. Berkeley, CA: University of California Press, 1991.

[24] Eun, Y. and Bang, H., "Cooperative control of multiple ucavs for suppression of enemy air defense," in *AIAA 3rd Unmanned Unlimited*, (Chicago, IL), Sep. 2004.

[25] Ferrari-Trecate, G., Egerstedt, M., Buffa, A., and Ji, M., "Laplacian sheep: A hybrid, stop-go policy for leader-based containment control," in *Hybrid Systems: Computation and Control*, pp. 212–226, Springer-Verlag, 2006.

[26] Ferrari-Trecate, G., Egerstedt, M., Buffa, A., and Ji, M., "Laplacian sheep: A hybrid, stop-go policy for leader-based containment control," in *Hybrid Systems: Computation and Control*, pp. 212–226, Springer-Verlag, 2006.

[27] Floreano, D. and Mattiussi, C., *Bio-Inspired Artificial Intelligence*. Cambridge, MA: The MIT Press, 2008.

[28] Freeman, R., Yang, P., and Lynch, K., "Stability and convergence properties of dynamic average consensus estimators," in *IEEE Conference on Decision and Control*, 2006.

[29] Funston, P., Mills, M., Biggs, H., and Richardson, P., "Hunting by male lions: ecological influences and socioecological implications," *Animal Behavior*, vol. 56, pp. 1333–1345, 1998.

[30] Gazi, V. and Passino, K., "A class of attraction/repulsion functions for stable swarm aggregations," *Int. Journal of Control*, vol. 77, pp. 1567–1579, Dec. 2004.

[31] Giraldeau, L.-A. and Caraco, T., *Social foraging theory*. Princeton, NJ: Princeton University Press, 2000.

[32] Grünbaum, D., Viscido, S., and Parrish, J., "Extracting interactive control algorithms from group dynamics of schooling fish," in *Proc. of Block Island Workshop on Cooperative Control* (Kumar, V., Leonard, N. E., and Morse, A. S., eds.), Springer-Verlag, 2004.

[33] Haque, M. and Egerstedt, M., "Decentralized formation selection mechanisms inspired by foraging bottlenose dolphins," in *Mathematical Theory of Networks and Systems*, (Blacksburg, VA), Jul. 2008.

[34] Haque, M., Rahmani, A., and Egerstedt, M., "Geometric foraging strategies in multi-agent systems based on biological models," in *Conference on Decision and Control*, (Atlanta, GA), Dec. 2010.

[35] Haque, M., A.R.Rahmani, and Egerstedt, M., "Biologically inspired coalition formation of multi-agent systems," in *Autonomous Agents and Multi-Agent Systems*, (Toronto, Canada), May 2010.

[36] Haque, M., A.R.Rahmani, and Egerstedt, M., "Biologically inspired confinement of multi-robot systems," *Submitted to Int. Journal of Bio-inspired Computation*, 2010.

[37] Haque, M., A.R.Rahmani, Egerstedt, M., and Yezzi, A., "Efficient foraging strategies in multi-agent systems through curve evolutions," *Submitted to Transactions on Automatic Control*, 2010.

[38] Haque, M., A.R.Rahmani, Egerstedt, M., and Yezzi, A., "Biologically motivated shape optimization of foraging fronts," in *Submitted to American Control Conference*, (San Francisco, CA), Jun. 2011.

[39] Haque, M. and Egerstedt, M., "Coalition formation in multi-agent systems based on bottlenose dolphin alliances," in *Proceedings of the American Control Conference*, (St. Louis, MO, USA), Jun. 2009.

[40] Haque, M., Egerstedt, M., and Martin, C., "First-order, networked control models of swarming silkworm moths," in *Proceedings of the American Control Conference*, Jun. 2008.

[41] Haque, M., Egerstedt, M., and Martin, C., "Sustainable group sizes for multi-agent search-and-patrol teams," in *Mathematical Theory of Networks and Systems*, (Budapest, Hungary), Jul. 2010.

[42] Haque, M., Rahmani, A., and Egerstedt, M., "A hybrid, multi-agent model of foraging bottlenose dolphins," in *3rd IFAC Conference on Analysis and Design of Hybrid Systems*, (Zaragoza, Spain), Sep. 2009.

[43] Heithaus, M. and Dill, L., "Food availability and tiger shark predation risk influence bottlenose dolphin habitat use," *Ecology*, vol. 83, no. 2, pp. 480–491, 2002.

[44] Henzinger, T., "The theory of hybrid automata," *Proceedings of the 11th Annual Symposium on Logic in Computer Science (LICS)*, pp. 278–292, 1996.

[45] Jadbabaie, A., Lin, J., and Morse, A., "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, pp. 988–1001, Jun. 2003.

[46] Ji, M. and Egerstedt, M., "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Transactions on Robotics*, vol. 23, pp. 693–703, Aug. 2007.

[47] Ji, M., Muhammad, A., and Egerstedt, M., "Leader-based multi-agent coordination: Controllability and optimal control," in *Proceedings of the American Control Conference*, pp. 1358–1363, 2006.

[48] Ketchpel, S., "Forming coalitions in the face of uncertain rewards," in *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pp. 414–419, MIT Press/AAAI Press, 1994.

[49] Kraus, S., Shehory, O., and Taase, G., "Coalition formation with uncertain heterogeneous information," in *AAMAS*, pp. 14–18, Jul. 2003.

[50] Krause, J., Hoare, D., Couzin, I., and Godin, J.-G., "Context-dependent group size choice in fish," *Animal Behaviour*, vol. 67, pp. 155–164, 2004.

[51] Krützen, M., Mann, J., Heithaus, M., Connor, R., Bejder, L., and Sherwin, W., "Cultural transmission of tool use in bottlenose dolphins," *Proc. Natl. Acad. Sci.*, vol. 102, pp. 8939–8943, Apr. 2005.

[52] Labella, T., Dorigo, M., and Deneubourg, J., "Self-organised task allocation in a group of robots," in *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems (DARS04)*, (Toulouse, France), Jun. 2004.

[53] Lankton, S., Nain, D., Yezzi, A., and Tannenbaum, A., "Hybrid geodesic region-based curve evolutions for image segmentation," *SPIE Medical Imaging*, 2007.

[54] Lee, S.-H., Pak, H., and Chon, T.-S., "Dynamics of prey-flock escaping behavior in response to predator's attack," *Journal of Theoretical Biology*, vol. 240, pp. 250–259, 2006.

[55] Lemmens, N., Jong, S., Tuyls, K., and Nowé, A., "Bee behaviour in multi-agent systems," in *Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning*, pp. 145–156, 2008.

[56] Lerman, K. and Galstyan, A., "Mathematical model of foraging in a group of robots: Effect of interference," *Autonomous Robots*, vol. 13, pp. 127–141, 2002.

[57] Liu, Y. and Passino, K., "Stable social foraging swarms in a noisy environment," *IEEE Trans. on Automatic Control*, vol. 49, pp. 30–44, Jan. 2004.

[58] Mann, J., Connor, R., Tyack, P., and Whitehead, H., *Cetacean Societies*. Chicago, IL: The University of Chicago Press, 2000.

[59] McNew, J., Klavins, E., and Egerstedt, M., "Solving coverage problems with embedded graph grammars," in *Hybrid Systems: Computation and Control*, pp. 413–427, Springer-Verlag, 2007.

[60] Mesbahi, M. and Egerstedt, M., *Graph theoretic methods in multiagent networks*. Princeton, NJ: Princeton University Press, 2010.

[61] Moon, S., Nabet, B., Leonard, N., Levin, S., and Kevrekidis, I., "Heterogeneous animal group models and their group-level alignment dynamics: An equation-free approach," *Journal of Theoretical Biology*, vol. 246, pp. 100–112, 2007.

[62] Muller, I., Kowalczyk, R., and Braun, P., "Towards agent-based coalition formation for service composition," in *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pp. 73–80, Dec. 2006.

[63] Murray, J., *Mathematical Biology I: An Introduction*. New York, NY: Springer, 2002.

[64] Nabet, B., Leonard, N., Couzin, I., and Levin, S., "Leadership in animal group motion: A bifurcation analysis," in *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, (Kyoto, Japan), 2006.

[65] Notarstefano, G., Egerstedt, M., and Haque, M., "Rendezvous with multiple, intermittent leaders," in *Conference on Decision and Control*, 2009.

[66] Notarstefano, G., Egerstedt, M., and Haque, M., "Containment in leader-follower networks with switching communication topologies," *Submitted to Automatica*, 2010.

[67] Olfati-Saber, R., "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, pp. 401–420, Mar. 2006.

[68] Olfati-Saber, R. and Murray, R., "Agreement problems in networks with directed graphs and switching topology," in *IEEE Conference on Decision and Control*, 2003.

[69] Osbourne, M., *An introduction to game theory*. New York, NY: Oxford University Press, 2004.

[70] Østergaard, E. H., Sukhatme, G. S., and Matarić, M. J., "Emergent bucket brigading," in *Proceedings of the fifth international conference on Autonomous agents*, (Montreal, Canada), 2001.

[71] Packer, C., Gilbert, D., Pusey, A., and O'Brien, S., "A molecular genetic analysis of kinship and cooperation in african lions," *Nature*, vol. 351, pp. 562–565, 1991.

[72] Packer, C. and Pusey, A., "Male takeovers and female reproductive parameters: A simulation of oestrous synchrony in lions (panthera leo)," *Animal Behavior*, vol. 31, pp. 334–340, 1983.

[73] Packer, C. and Pusey, A., "Divided we fall: Cooperation among lions," *Scientific American Magazine*, May 1997.

[74] Pratt, S., Sumpter, D., Mallon, E., and Franks, N., "An agent-based model of collective nest choice by the ant temnothorax curvispinosus," *Animal Behaviour*, vol. 70, pp. 1023–1036, 2005.

[75] Pryor, K. and Norris, K., *Dolphin Societies*. Berkely, CA: University of California Press, 1998.

[76] Rahmani, A., Ji, M., Mesbahi, M., and Egerstedt, M., "Controllability of multi-agent systems from a graph-theoretic perspective," *SIAM Journal on Control and Optimization*, vol. 48, pp. 162–186, Feb. 2009.

[77] Rasmussen, S., Mitchell, J., and Allen, K., "Simulation framework for moving cooperative control algorithms from conception to assessment," in *AIAA Modeling and Simulation Technologies Conference and Exhibit*, (Providence, RI), Aug. 2004.

[78] Ren, W. and Beard, R., "Consensus of information under dynamically changing interaction topologies," in *Proceedings of the American Control Conference*, pp. 4939–4944, 2004.

[79] Reynolds, C., "Flocks, herds, and schools: A distributed behavioral model," *Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.

[80] Rhodes, J. and Holder, G., *Concept for Future Naval Mine Countermeasures in Littoral Power Projection.* 1998.

[81] Roth, G. and Dicke, U., "Evolution of the brain and intelligence," *TRENDS in Cognitive Sciences*, vol. 9, pp. 250–257, May 2005.

[82] Sandholm, T. and Lesser, V., "Coalitions among computationally bounded agents," *Artificial Intelligence*, vol. 94, no. 1, pp. 99–137, 1997.

[83] Sariel, S. and Balch, T., "A distributed multi-robot cooperation framework for real time task achievement," in *Distributed Autonomous Robotic Systems (DARS) 7*, pp. 187–196, 2006.

[84] Scheel, D., "Profitability, encounter rates and the prey choice of african lions," *Behavioral Ecology*, vol. 4, pp. 90–97, 1993.

[85] Scheel, D. and Packer, C., "Group hunting behaviour of lions: a search for cooperation," *Animal Behavior*, vol. 41, pp. 697–709, 1991.

[86] Schusterman, R., Thomas, J., and Wood, F., *Dolphin Cognition and Behavior: A Comparative Approach.* Hillsdale, NJ: Lawrence Erlbaum associates, Publishers, 1986.

[87] Sen, S. and Dutta, P., "Searching for optimal coalition structures," in *Proceedings of the Fourth International Conference on MultiAgent Systems*, pp. 287–292, Jul. 2000.

[88] Shehory, O. and Kraus, S., "Coalition formation among autonomous agents: Strategies and complexity," *Artificial Intelligence*, pp. 57–72, 1995.

[89] Shehory, O. and Kraus, S., "Methods for task allocation via agent coalition formation," *Artificial Intelligence*, vol. 101, pp. 165–200, 1998.

[90] Shell, D. and Matarić, M., "On foraging strategies for large-scale multi-robot systems," in *Intl. Conf. on Intelligent Robots and Systems*, pp. 2717–2723, 2006.

[91] Shi, Y. and Karl, W., "A fast level set method without solving pdes," in *Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 97–100, Dec. 2005.

[92] Sichman, J., Conte, R., Demazeau, Y., and Castelfranchi, C., "A social reasoning mechanism based on dependence networks," in *16th Eur. Conf. Artificial Intelligence*, pp. 188–192, 1994.

[93] Slijper, E., *Whales and Dolphins.* Ann Arbor, MI: The University of Michigan Press, 1976.

[94] Smith, B., Egerstedt, M., and Howard, A., "Automatic generation of persistent formations for multi-agent networks under range constraints," *Mobile Networks and Applications Journal*, vol. 14, pp. 322–335, Jun. 2009.

[95] Smith, B., Howard, A., McNew, J.-M., Wang, J., and Egerstedt, M., "Multi-robot deployment and coordination with embedded graph grammars," *Autonomous Robots*, vol. 26, pp. 79–98, Jan. 2009.

[96] Tsui, P. and Hu, H., "A framework for multi-robot foraging over the internet," in *Proceedings of IEEE International Conference on Industrial Technology*, (Bangkok, Thailand), 2002.

[97] Yeung, D.-Y. and Bekey, G., "A decentralized approach to the motion planning problem for multiple mobile robots," *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, pp. 1779–1784, Mar. 1987.